

# ***Institut Universitaire des Sciences (IUS)***

**Faculté des Sciences et Technologies  
(FST)**

## **RAPPORT SUR LE TRAVAIL DE LABORATOIRE N° 8**

*Cours : Système d'exploitation*

Soumis au Chargé de cours : **Ismael SAINT AMOUR**

Niveau L3

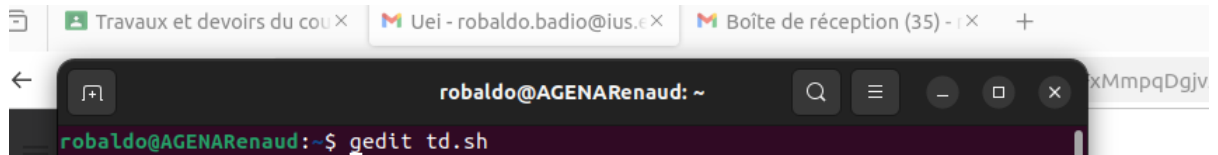
Préparé par : **Robaldo BADIO**

*Date : Le 27 / 01 / 2025*

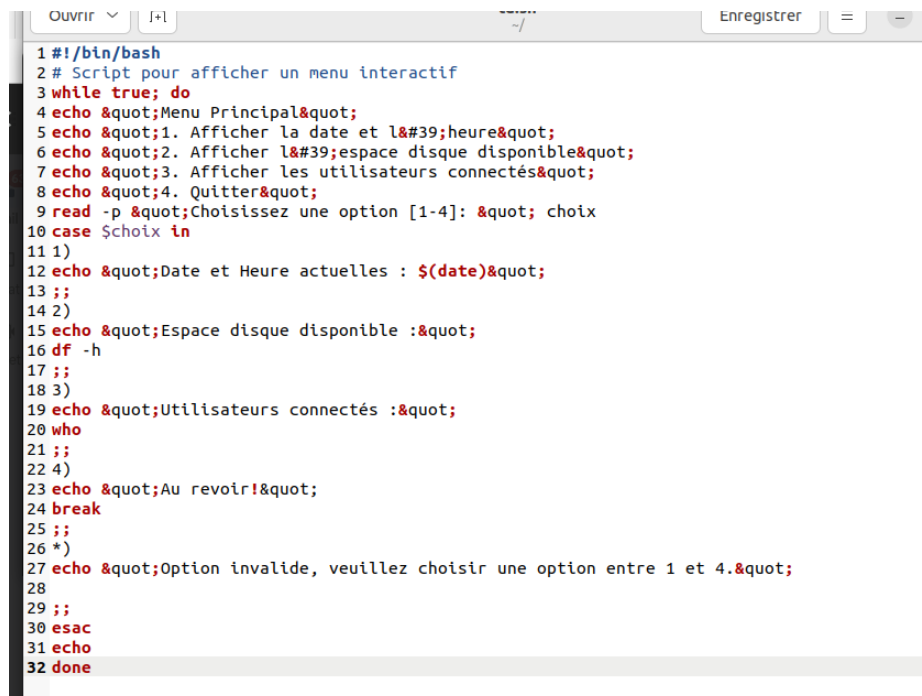
## Exécution du TD

### Travaux Dirigés

1. Créez un script qui affiche les multiples de 5 entre 5 et 100 ;



2. Créez un script avec une boucle qui affiche un menu interactif à l'utilisateur ;



3. Créez un script qui permet d'effectuer des opérations sur les variables ;

```
Editeur de texte jan 27 13:49
Ouvrir  td.sh  Enregistrer

1 #!/bin/bash
2 # Script pour effectuer des opérations sur les variables
3 while true; do
4 echo &quot;Menu des opérations&quot;;
5 echo &quot;1. Addition&quot;;
6 echo &quot;2. Soustraction&quot;;
7 echo &quot;3. Multiplication&quot;;
8 echo &quot;4. Division&quot;;
9 echo &quot;5. Quitter&quot;;
10 read -p &quot;Choisissez une option [1-5]: &quot; choix
11 if [ &quot;$choix&quot; -eq 5 ]; then
12 echo &quot;Au revoir!&quot;;
13 break
14 fi
15 read -p &quot;Entrez la première variable: &quot; var1
16 read -p &quot;Entrez la deuxième variable: &quot; var2
17 case $choix in
18 1)
19 result=$((var1 + var2))
20 echo &quot;Résultat de l'addition : $var1 + $var2 = $result&quot;;
21 ;;
22 2)
23 result=$((var1 - var2))
24 echo &quot;Résultat de la soustraction : $var1 - $var2 = $result&quot;;
25 ;;
26 3)
27 result=$((var1 * var2))
28 echo &quot;Résultat de la multiplication : $var1 * $var2 = $result&quot;;
29 ;;
30 4)
31 if [ &quot;$var2&quot; -ne 0 ]; then
32 result=$((var1 / var2))
33 echo &quot;Résultat de la division : $var1 / $var2 = $result&quot;;
34 else
35 echo &quot;Erreur : Division par zéro n'est pas possible.&quot;;
36 fi
37 fi
```

```
36 echo &quot;Erreur : Division par zéro n'est pas possible.&quot;;
37 fi
38 ;;
39 *)
40 echo &quot;Option invalide, veuillez choisir une option entre 1 et 5.&quot;;
41 ;;
42 esac
43 echo
44 done

sh  Largeur des tabulations : 8  Lig 44, Col 5  INS
```

4. Créez un script qui utilise une boucle until pour demander à l'utilisateur de fournir un fichier existant et un mot à rechercher ;

```
td.sh
Ouvrir  Enregistrer

1 #!/bin/bash
2 # Script pour demander un fichier existant et un mot à rechercher
3 fichier=&quot;&quot;;
4 mot=&quot;&quot;;
5 # Boucle jusqu'à ce qu'un fichier existant soit fourni
6 until [ -f &quot;$fichier&quot; ]; do
7 read -p &quot;Veuillez fournir le chemin du fichier existant : &quot; fichier
8 if [ ! -f &quot;$fichier&quot; ]; then
9 echo &quot;Le fichier $fichier n'existe pas. Veuillez essayer de nouveau.&quot;;
10 fi
11 done
12 # Demander un mot à rechercher
13 read -p &quot;Veuillez fournir le mot à rechercher : &quot; mot
14 # Rechercher le mot dans le fichier
15 echo &quot;Recherche du mot $mot dans le fichier $fichier...&quot;;
16 grep -n &quot;$mot&quot; &quot;$fichier&quot;;
17 # Afficher un message si le mot n'est pas trouvé
18 if [ $? -ne 0 ]; then
19 echo &quot;Le mot $mot n'a pas été trouvé dans le fichier $fichier.&quot;;
20 else
21 echo &quot;Le mot $mot a été trouvé dans le fichier $fichier.&quot;;
22 fi
```

5. Créez un script qui demande un mot à l'utilisateur jusqu'à ce qu'il tape "Au revoir!" ;

```
1 #!/bin/bash
2 # Script qui demande un mot à l'utilisateur jusqu'à ce qu'il tape "Au revoir!"
3 mot=""
4 until [ "$mot" = "Au revoir!" ]; do
5 read -p "Veuillez entrer un mot (taper 'Au revoir!' pour quitter) : " mot
6 if [ "$mot" != "Au revoir!" ]; then
7 echo "Vous avez entré : $mot"
8 fi
9 done
10 echo "Au revoir!"
```

6. Créez un script qui génère la table de multiplication ou division pour un nombre donné ;

```
1 #!/bin/bash
2 # Script pour générer une table de multiplication ou de division pour un nombre donné
3 read -p "Choisissez l'opération (1 pour multiplication, 2 pour division) : " operation
4 read -p "Entrez le nombre pour lequel vous voulez générer la table : " nombre
5 if [ "$operation" = "1" ]; then
6 echo "Table de multiplication pour $nombre"
7 for i in {1..10}; do
8 result=$((nombre * i))
9 echo "$nombre x $i = $result"
10 done
11 elif [ "$operation" = "2" ]; then
12 echo "Table de division pour $nombre"
13 for i in {1..10}; do
14 if [ $i -ne 0 ]; then
15 result=$((echo "scale=2; $nombre / $i" | bc))
16 echo "$nombre / $i = $result"
17 fi
18 done
19 else
20 echo "Opération invalide. Veuillez choisir 1 pour multiplication ou 2 pour division."
21 fi
```

7. Créez un Script pour Générer un Récit ;

```
1 #!/bin/bash
2 # Script pour générer un récit interactif
3 # Demander les informations à l'utilisateur
4 read -p "Entrez un nom : " nom
5 read -p "Entrez un lieu : " lieu
6 read -p "Entrez un objet : " objet
7 read -p "Entrez une action (verbe) : " action
8 # Générer le récit
9 echo ""
10 echo "Voici votre récit :"
11 echo "Il était une fois, dans un lieu nommé $lieu, un personnage nommé $nom."
12 echo "Un jour, $nom a trouvé un $objet mystérieux."
13 echo "Avec beaucoup de courage, $nom a décidé de $action le $objet."
14 echo "Cette aventure a changé la vie de $nom pour toujours."
15 echo ""
16 # Fin du script
17 echo "Merci d'avoir créé ce récit avec nous !"
```

## 8. Créez un Script Collecter les Informations d'étudiant ;

```
1 #!/bin/bash
2
3 # Script pour collecter les informations des étudiants
4
5 # Fichier de sortie pour stocker les informations
6 fichier_etudiants="informations_etudiants.txt"
7
8 # Fonction pour ajouter les informations d'un étudiant au fichier
9 ajouter_etudiant() {
10 echo "Nom complet: $nom_complet" >> $fichier_etudiants
11 echo "Date de naissance: $date_naissance" >> $fichier_etudiants
12 echo "Âge: $age" >> $fichier_etudiants
13 echo "Adresse: $adresse" >> $fichier_etudiants
14 echo "Numéro de téléphone: $tel" >> $fichier_etudiants
15 echo "Email: $email" >> $fichier_etudiants
16 echo "Sexe: $sexe" >> $fichier_etudiants
17 echo "-----" >> $fichier_etudiants
18 }
19
20 # Boucle pour collecter les informations des étudiants
21 while true; do
22 read -p "Entrez le nom complet de l'étudiant (ou tapez # pour quitter) : " nom_complet
23 if [ $nom_complet == # ]; then
24 echo "Fin de la collecte des informations."
25 break
26 fi
27
28 read -p "Entrez la date de naissance (JJ/MM/AAAA) : " date_naissance
29 read -p "Entrez l'âge de l'étudiant : " age
30 read -p "Entrez l'adresse : " adresse
31 read -p "Entrez le numéro de téléphone : " tel
32
33 # Demander l'email et vérifier qu'il contient le caractère @
34 while true; do
35 read -p "Entrez votre adresse email : " email
36 if [ $email =~ @ ]; then
37 break
38 else
39 echo "L'adresse email doit contenir le caractère @. Essayez encore."
40 fi
41 done
42
43 # Choisir le sexe
44 echo "Choisissez votre sexe : "
45 echo "1. Homme"
46 echo "2. Femme"
47 echo "3. Autre"
48
49 while true; do
50 read -p "Votre choix (1/2/3) : " sexe
51 case $sexe in
52 1) sexe=Homme; break ;;
53 2) sexe=Femme; break ;;
54 3) sexe=Autre; break ;;
55 *) echo "Choix invalide. Veuillez entrer 1, 2 ou 3." ;;
56 esac
57 done
58
59 # Ajouter les informations de l'étudiant au fichier
60 ajouter_etudiant
61
62 # Afficher les informations collectées
63 echo "Informations collectées : "
64 echo "Nom complet : $nom_complet"
65 echo "Date de naissance : $date_naissance"
66 echo "Âge : $age"
67 echo "Adresse : $adresse"
68 echo "Numéro de téléphone : $tel"
69 echo "Email : $email"
70 echo "Sexe : $sexe"
```

```
72 echo " "
73 done
74
75 # Fin du script
76 echo "Toutes les informations ont été collectées et enregistrées dans $fichier_etudiants."
```

9. Créez un script qui demande à l'utilisateur de fournir le nom du dossier, demande à l'utilisateur combien de fichiers texte il souhaite créer, crée les fichiers texte et les ajoute au dossier, liste tous les fichiers créés.



```
1#!/bin/bash
2# Demande à l'utilisateur de fournir le nom du dossier
3read -p "Entrez le nom du dossier : " dossier
4# Crée le dossier s'il n'existe pas déjà
5if [ ! -d "$dossier" ]; then
6mkdir -p "$dossier";
7echo "Le dossier '$dossier' a été créé.";
8else
9echo "Le dossier '$dossier' existe déjà.";
10fi
11# Demande à l'utilisateur combien de fichiers texte il souhaite créer
12read -p "Combien de fichiers texte souhaitez-vous créer ? " nombre_fichiers
13# Boucle pour créer les fichiers texte
14for ((i=1; i<=nombre_fichiers; i++)); do
15fichier="$dossier/fichier_$i.txt";
16touch "$fichier";
17echo "Fichier '$fichier' créé.";
18done
19# Liste tous les fichiers créés
20echo "";
21echo "Liste des fichiers créés dans le dossier '$dossier' :";
22ls -l "$dossier";
```

En conclusion, ce TD me permet d'approfondir mes connaissances sur l'écriture de scripts Shell de base et comment utiliser des variables.

Il me permet aussi de comprendre et de faire des pratiques sur quelques concepts à savoir : la structure de base d'un script shell, comment manipuler des variables dans un script, l'utilisation des conditions (if, case), comment implémenter des boucles (for, while) et l'Automatisation des tâches simples avec des scripts shell.