

# TD4 – Langages de script

Abdallah Ammar

12 janvier 2026

## Objectifs du TD

Ce TD a pour objectif de vous initier à l'écriture de scripts bash. À l'issue de la séance, vous devrez être capables de :

- créer et exécuter un script bash ;
- utiliser des variables ;
- passer des paramètres à un script ;
- tester des conditions simples ;
- automatiser des commandes déjà connues.

Il est fortement recommandé de taper les commandes et le code à la main afin de bien comprendre le fonctionnement des scripts.

## 1 Qu'est-ce qu'un script bash ?

Un script bash est un fichier texte contenant une suite de commandes qui seront exécutées automatiquement par le shell.

La première ligne d'un script bash est appelée *shebang* : `#!/bin/bash`

## 2 Créer et exécuter un premier script

### Travail à faire

1. Créez un fichier nommé `script.sh`.
2. Ajoutez la ligne `#!/bin/bash`.
3. Affichez un message à l'écran.
4. Rendez le script exécutable.
5. Exécutez le script.

### Commandes utiles

```
$ chmod +x script.sh $ ./script.sh
```

## 3 Variables

Les variables permettent de stocker des informations dans un script.

## Travail à faire

1. Créez une variable contenant votre prénom.
2. Affichez la valeur de cette variable.
3. Modifiez la variable.
4. Réaffichez-la.

## Exemple

```
prenom="Alice" echo $prenom
```

## 4 Paramètres d'un script

Un script peut recevoir des paramètres lors de son exécution.

## Travail à faire

1. Modifiez votre script pour afficher le premier paramètre.
2. Testez le script avec différents paramètres.
3. Que contient la variable \$0 ?

## Exemple

```
echo $1
```

## 5 Tests et conditions simples

Les structures conditionnelles permettent d'exécuter du code en fonction d'une condition.

## Travail à faire

1. Si aucun paramètre n'est fourni, affichez un message d'erreur.
2. Sinon, affichez le paramètre fourni.

## Structure générale

```
if [ condition ]; then commandes fi
```

## 6 Tester l'existence d'un fichier

## Travail à faire

Écrivez un script qui :

- prend un nom de fichier en paramètre;
- affiche "Fichier trouvé" s'il existe;
- affiche "Fichier inexistant" sinon.

## Tests utiles

```
-f fichier    (teste si c'est un fichier) -d dossier    (teste si c'est un  
répertoire)
```

## 7 Mini-script d'automatisation

### Travail à faire

Écrivez un script qui :

- prend un répertoire en paramètre ;
- vérifie qu'il existe ;
- affiche le nombre de fichiers qu'il contient.

### Indications

Vous pouvez utiliser les commandes suivantes :

- `ls`
- `wc -l`

### Remarques

- Un script est simplement une suite de commandes déjà connues.
- Les erreurs sont normales lors de l'écriture d'un script.
- Lisez attentivement les messages d'erreur.

### Pour aller plus loin – Exercice ludique (optionnel)

Écrivez un script bash affichant un message personnalisé à l'utilisateur, en utilisant une variable et un paramètre.

Par exemple :

- `./bonjour.sh Alice`
- affiche : `Bonjour Alice !`