

TD11 – Langages de script

Abdallah Ammar

12 janvier 2026

Objectifs du TD

Ce TD a pour objectif d'apprendre à écrire des scripts bash plus lisibles, plus structurés et plus faciles à maintenir. À l'issue de la séance, vous devrez être capables de :

- découper un script en plusieurs fonctions ;
- comprendre le rôle des variables dans un script ;
- organiser un script en parties claires ;
- ajouter une aide pour l'utilisateur.

Dans ce TD, l'objectif n'est pas d'écrire plus de code, mais d'écrire du code plus clair et mieux organisé.

1 Pourquoi structurer un script ?

Jusqu'à présent, vous avez écrit des scripts contenant une suite de commandes. Lorsque les scripts deviennent plus longs, il devient difficile :

- de comprendre ce qu'ils font ;
- de les modifier ;
- de corriger des erreurs.

Structurer un script permet de le rendre plus lisible et plus robuste.

2 Découper un script en fonctions

Une fonction permet de regrouper des commandes ayant un rôle précis.

Travail à faire

Reprenez un script écrit lors d'un TD précédent (TD6 ou TD7 par exemple) et identifiez les différentes parties du script.

Par exemple :

- vérification des paramètres ;
- analyse d'un fichier ;
- affichage des résultats.

Créez une fonction pour chacune de ces parties.

3 Définir et utiliser des fonctions

Structure générale

```
nom_fonction() { commandes }
```

Une fonction peut ensuite être appelée simplement par son nom.

Travail à faire

1. Créez une fonction `check_args` qui vérifie les paramètres.
2. Créez une fonction `analyze` qui effectue le traitement principal.
3. Créez une fonction `print_result` qui affiche les résultats.
4. Appelez ces fonctions dans la partie principale du script.

4 Variables et portée

Les variables définies dans un script sont accessibles dans les fonctions.

Travail à faire

1. Définissez une variable au début du script.
2. Modifiez cette variable dans une fonction.
3. Affichez la variable après l'appel de la fonction.

Question

Que constatez-vous ? Expliquez pourquoi la valeur de la variable a changé.

5 Ajouter une aide au script

Un script bien écrit doit expliquer comment il s'utilise.

Travail à faire

Ajoutez une option `-h` ou `-help` à votre script. Lorsque cette option est fournie, le script doit afficher :

- une description du script ;
- les paramètres attendus ;
- un exemple d'utilisation.

Indication

Vous pouvez utiliser une structure conditionnelle du type :

```
if [ "$1" = "-h" ]; then echo "Aide du script" exit 0 fi
```

6 Organisation finale du script

Votre script final doit être organisé de la manière suivante :

- commentaires en début de fichier ;
- définition des variables ;
- définition des fonctions ;
- partie principale (appels des fonctions).

Travail à faire

Réorganisez votre script pour qu'il respecte cette structure.

Remarques

- Un script lisible est plus important qu'un script court.
- Les fonctions permettent d'éviter les répétitions.
- Ce TD prépare l'écriture de scripts plus complexes.