

Pratique des machines

TP5 : pratiquer les notions

am@up8.edu

Octobre 2025

Dans ce TP :

- écriture de scripts bash
- redirection avec [|](#)

→ **Avant de commencer** : créer un répertoire TP5 dans le répertoire dédié à ce cours sur votre session.

1 Exercice 1 : manipulation d'une base de données textuelle

1. Placez votre répertoire de travail répertoire TP5;
2. Créez un fichier TP5_exercice1.sh dans lequel vous consignez toutes les commandes (fonctionnelles) que vous utilisez. À la fin de la séance, le script TP5_exercice1.sh doit vous permettre de reproduire les commandes de réponses aux différentes questions. Si vous oubliez certaines commandes, vous pouvez utiliser l'historique qui se trouve dans le fichier `~.bash_history`.
3. Télécharger un fichier depuis le terminal

Copier/coller un lien depuis un pdf

Pour copier un lien, faites clic-droit dessus puis cliquez sur copier l'adresse du lien (ou équivalent). Vous pouvez ensuite le coller dans le terminal en faisant simultanément au clavier les touches `ctrl`, `maj` et `v`.

Vous allez télécharger le contenu disponible au bout de ce lien :

<https://www.data.gouv.fr/fr/datasets/r/b363e051-9649-4879-ae78-71ef227d0cc5> grâce à la commande **wget**;

```
$ wget https://www.data.gouv.fr/fr/datasets/r/b363e051-9649-4879-ae78-71ef227d0cc5 # (ATTENTION ne copiez-collez pas ici, suivez la procédure au dessus)
```

4. Renommez le fichier téléchargé et donnez lui le nom : **base_de_donnees.csv**;

```
$ mv b363e051-9649-4879-ae78-71ef227d0cc5 base_de_donnees.csv
```

5. Exécutez la commande suivante pour vous faire une idée de ce que contient le fichier. Vous pouvez naviguer dans le fichier « page par page » en utilisant les touches « f » (*forward*, vers l'avant) et « b » (*backward*, vers l'arrière) et tapez « q » lorsque vous avez fini :

```
$ less base_de_donnees.csv
```

6. Utilisez la commande **head** (la « tête ») et ses options pour récupérer uniquement la première ligne de ce fichier, et stockez-la dans un fichier nommé **en_tete.csv** (vous devez utiliser la redirection avec l'opérateur >);

Recommandations

- (a) de manière générale, n'utilisez pas de caractères accentués, ni d'espaces, ni de caractères spéciaux (par exemple « { ») dans vos noms de fichiers ou de répertoires;
- (b) vérifiez toujours l'extension des fichiers et répertoires que vous créez ou manipulez.

7. Grâce à la commande **tail** (la « queue »), récupérez toutes les lignes sauf la première et stockez-les dans un fichier nommé **corps.csv** (en utilisant la redirection). Vous aurez sans doute besoin de connaître le nombre de lignes contenues dans le fichier au préalable.
8. commande **sort**

```
$ sort [OPTION]... [FILE...]
```

La commande **sort** permet de trier un fichier.

L'option **-k** permet d'indiquer selon quelle "clé" (numéro de colonne ici) il faut trier.

L'option **-t** permet d'indiquer quel est le séparateur de colonnes.

Par exemple avec un fichier **.txt** qui contient :

```
1 Benjamin L1A 1994
2 Alice L1B 1991
```

la commande :

```
$ sort -t " " -k 4 fichier.txt
```

trie selon la 4^e colonne en considérant l'espace comme le séparateur.

En utilisant la commande **sort**, triez les entrées de **corps.csv** par ordre de numéro de département, et trouvez un moyen de ne récupérer que les lignes concernant les académies dont le numéro de département est inférieur à 30, stockez les dans un fichier **dep_inf_30.csv**.

9. Extrayez les entrées des académies en zone A grâce à la commande **grep** et stockez-les dans un fichier **zone_A.csv**

10. Commande **comm**

Exécutez :

```
$ comm dep_inf_30.csv zone_A.csv
```

comm produit une comparaison des deux fichiers passés en paramètres et renvoie sur trois colonnes : les lignes présentes dans le premier fichier seulement, les lignes présentes dans le second fichier seulement, et les lignes présentes dans les deux fichiers.

Exécutez :

```
$ comm -1 -3 dep_inf_30.csv zone_A.csv
```

Les options **-1** et **-3** permettent de n'afficher que les lignes présentes *seulement dans le second fichier*. Cela fonctionne-t-il ?

Consultez la documentation de **comm** et faites les modifications nécessaires afin de stocker dans un fichier *les lignes des académies dont le département est inférieur à 30 ne se trouvant pas en zone A*.

11. En utilisant la commande **cut** et à la commande **tr**, extrayez la liste des toponymes (villes, îles, agglomérations) présents dans la première colonne et classez-les par ordre alphabétiques. Supprimez les doublons et stockez la liste des toponymes dans un fichier **toponymes.txt**.
12. Extrayez de **corps.csv** la liste des lignes qui contiennent au moins un champ vide et stockez les dans un fichier dont le nom vous semble approprié.
13. commande **paste** :
 - En utilisant **echo** et la redirection (**>**), créez en une instruction un fichier **zones.txt** contenant le texte suivant :

```
1 $ Zone A
```

2 \$ Zone B

3 \$ Zone C

- en utilisant 3 fois **grep** et **wc**, ainsi que les opérateurs **>** et **>>** stockez dans un fichier **compte.txt** le nombre d'entrées correspondant à chaque zone.
- utilisez une commande et la redirection (**>**) pour créer un nouveau fichier **zone_compte.txt** dans lequel les zones et le nombre d'académies concernées apparaissent *côte à côte*. Regarder les documentations des commandes **join** et **paste** afin de choisir la plus adaptée.

2 Exercice 2 : écrire un petit script

Les étapes ci-dessous n'ont pas toujours de lien entre-elles. Efforcez-vous de bien comprendre la consigne avant de choisir la commande appropriée.

Créez un script `TP5_exercice2.sh` qui :

1. crée un dossier `env` dans le dossier de téléchargement de votre home (donc Downloads ou Téléchargements)
2. se déplace dans ce dossier (`~/Téléchargements`(ou `Downloads`))/`env` devient votre répertoire de travail)
3. copie le fichier `/etc/environment` dans le répertoire de travail
4. imprime le contenu de tous les fichiers du répertoire de travail dans le terminal
5. affiche le message "Mon ordinateur a tant de processeurs : ", sans retour à la ligne
6. effectue l'enchaînement de commandes suivant (utiliser l'opérateur `|`) :
 - (a) liste le contenu du dossier `cpu` situé dans `/dev`
 - (b) filtre les noms de fichier ne contenant que des chiffres avec `grep -E "[0-9]+"`
 - (c) compte le nombre de lignes
 - (d) affiche le nombre obtenu sur la sortie standard
7. écrit le nombre obtenu dans un fichier `nb_process.txt`