

TP n°8

Année 2024-2025 – S1

Programmation shell

Important : Pensez à commenter vos scripts, pour vous rappeler ce que vous avez fait quand vous les reprendrez plus tard !

Ne bloquez pas sur un script. Si besoin, passez au suivant pour vous changer les idées, puis revenez sur le précédent. Vous pouvez également demander de l'aide à votre enseignant !

Vous pouvez rédiger vos scripts avec l'éditeur « nano » (cf diapo 66) ou l'éditeur de votre choix.

Rechercher des informations sur les villes de France

Le script réalisé au cours de ce TP a pour but de permettre la recherche d'informations sur les villes de France.

Créez une section « Paramètres » (avec un commentaire, cf. corrigés des TP précédents) au début de votre fichier pour y consigner les informations nécessaires dans des variables que vous appellerez au besoin dans votre script.

1. Récupérer les données des villes

L'état fournit la liste complète des villes de France avec un certain nombre d'informations (code insee, coordonnées GPS, département, etc.) à l'adresse suivante :

<https://www.data.gouv.fr/fr/datasets/villes-de-france/>

Vous pouvez notamment récupérer cette liste au format CSV grâce à l'URL suivante :

<https://www.data.gouv.fr/fr/datasets/r/51606633-fb13-4820-b795-9a2a575a72f1>

Ce fichier comporte les colonnes suivantes, dans l'ordre, séparées par des virgules :

| | |
|------------|---------------------|
| insee_code | longitude |
| city_code | department_name |
| zip_code | department_number |
| label | region_name |
| latitude | region_geojson_name |

- Dans la section paramètres, indiquez une variable contenant l'URL de la liste des villes, et une autre variable contenant le nom de fichier avec son chemin d'accès à utiliser pour télécharger la liste.
- Créez une fonction qui permet de télécharger la liste des villes sous le nom de fichier prévu à cet effet, et qui renvoie vrai uniquement si le téléchargement a réussi.

Indices : Diapos 64/65 et 243 à 248

2. Menu de choix

Créez une fonction qui va :

- Afficher un menu avec les choix suivants :
 - 1) Localiser une ville
 - 2) Lister les villes correspondant à un code postal
- Demander à l'utilisateur quel choix il souhaite faire et enregistrer ce choix dans une variable (1 ou 2)

Indices : Diapos 191 et 197 à 200

3. Programme principal

A la fin de votre script (pour rappel, les fonctions doivent être codées AVANT leur appel), appelez la fonction permettant de télécharger la liste des villes, et si le téléchargement réussit, appelez la fonction de menu.

Si le téléchargement échoue, affichez un message d'erreur.

4. Quitter proprement le programme

Créez une fonction qui permet de supprimer les éventuels fichiers créés pendant l'exécution du programme, puis qui quitte le programme en renvoyant le code de retour reçu en 1^{er} paramètre de la fonction.

Utilisez cette fonction pour quitter le programme si le téléchargement échoue.

5. Vérifier la saisie

Ajoutez une fonction qui prend une valeur en paramètre et qui vérifie s'il s'agit d'un nombre ou non. Si oui, la fonction doit renvoyer vrai, si non, la fonction doit renvoyer faux.

Utilisez cette fonction dans la fonction de menu pour vérifier si le choix de l'utilisateur est un nombre ou non.

Si oui, vérifiez si le nombre correspond à l'un des choix possibles.

Si le choix n'est pas un nombre, ou que ce n'est pas un choix valide, quittez le programme avec la fonction créée à cet effet en affichant un message d'erreur.

Indices : Diapos 221 ou 234, diapo 231

6. Localiser une ville

Créez une fonction qui prend en paramètre un nom de ville.

Recherchez (en ignorant la casse) et affichez les coordonnées GPS de la ville en question dans le fichier de villes.

L'affichage doit ressembler à cela :

```
Coordonnées GPS de belfort (Latitude,Longitude) : 47.64696911,47.64696911
```

Pour y arriver, vous devez passer par deux étapes :

- Tout d'abord, filtrez les colonnes pour ne conserver que le nom de la ville, la latitude et la longitude, filtrez le résultat pour ne garder que les lignes contenant la ville souhaitée, et enregistrez le résultat final dans une variable.

Votre variable devrait alors contenir la valeur suivante (pour Belfort) :

```
belfort,47.64696911,47.64696911
```

- Si la variable est vide, c'est qu'il n'y a pas de résultat.
Si elle contient quelque chose, refaites un filtrage du contenu de la variable pour ne récupérer que la 1^{ère}, la 2^e ou la 3^e colonne en fonction de ce que vous souhaitez afficher.

Indices : Diapos 82, 85/86, 87, 194 à 196

Dans la fonction de menu, si l'utilisateur choisit l'option 1, demandez-lui de saisir le nom d'une ville, puis lancez la fonction que vous venez de créer en lui donnant le nom de la ville saisi par l'utilisateur en paramètre, puis quittez le programme avec la fonction adéquate.

7. Lister les villes correspondant à un code postal

Créez une fonction qui prend en paramètre un code postal.

La fonction devra afficher la liste des villes dont le code postal correspond à celui transmis en paramètre.

La méthode à suivre est similaire à celle de la question 6, en utilisant en plus une boucle pour lister toutes les villes.

Attention à ne pas confondre code INSEE et code postal ! (zip_code dans le fichier)

Indices :

- Il est possible d'effectuer une boucle for avec un saut de ligne comme seul séparateur (pour les noms de ville contenant des espaces). Pour cela, ajoutez la ligne suivante juste avant votre for :

```
IFS=$'\n'
```


et ajoutez la ligne suivante juste après la boucle for pour annuler le réglage :

```
unset IFS
```
- Pour récupérer la sortie standard d'une commande dans une variable en conservant les sauts de lignes, il est nécessaire de mettre le \$() entre guillemets.

Exemple :

```
listeFichiers="$(ls -l)"
```

Même chose pour afficher une variable contenant des sauts de ligne :

```
echo "${listeFichiers}"
```

8. Quelques améliorations

- Dans la fonction de localisation d'une ville, affichez en plus le département et la région, ainsi qu'une URL pour effectuer un trajet jusqu'à cette ville.

Pour l'URL, utilisez :

`https://www.google.com/maps/dir/***LATITUDE***,***LONGITUDE***`

en remplaçant « ***LATITUDE*** » par la latitude de la ville et

« ***LONGITUDE*** » par la longitude de la ville (attention à ne pas oublier la virgule entre les deux valeurs !)

- Relancez le menu après l'affichage d'un résultat au lieu de quitter le programme. Ajoutez une option « 9) Quitter le programme » qui permettra de réellement quitter le programme (toujours en utilisant la fonction adéquate)
- Cherchez dans le man de wget comment faire en sorte de n'afficher que la barre de progression du téléchargement, sans le reste des détails du téléchargement.

Fin du TP :

Pensez à sauvegarder vos scripts sur une clef USB ou en vous les envoyant par mail !

Si vous vous envoyez vos fichiers par mail avec le webmail de l'IUT, renommez d'abord vos fichiers en .txt, le webmail n'accepte pas d'envoyer un mail avec des .sh en pièce jointe.

A la fin du TP, il faut réinitialiser les machines à leur état d'origine. Pour ce faire :

- Attendez que l'enseignant vous indique que la restauration peut être effectuée.
- Redémarrez alors votre machine.

- Dès que l'écran « Dell » s'affiche, tapotez rapidement la touche F12 jusqu'à ce que la mention « Preparing boot menu » apparaisse en haut à droite de l'écran.
- Après quelques instants, un menu apparait. Choisissez avec les flèches du clavier l'option « Onboard NIC (IPv4) », puis validez avec Entrer.
- Enfin, attendez quelques dizaines de secondes (maximum) pour vérifier que la machine démarre bien sur le système de restauration (le mot « FOG » devrait apparaitre en gros sur l'écran).