

Système - TP noté

Le but de ce TP est d'écrire un programme qui prend en entrée un site web sous forme de fichiers .html, et produit en sortie des fichiers au format CSV contenant toutes les données présentes sur le site.

Joint à ce sujet, vous trouverez un dossier "site" contenant tous les fichiers du site web. Vos scripts doivent être capables de fonctionner avec un site ayant la même structure mais contenant des données différentes. Ne présumez donc pas que le fichier `ue_liftruc.html` existe, même s'il est bien présent dans le site fourni.

Le site web est celui qui a été produit grâce aux scripts du TP3. On cherche maintenant à faire l'opération inverse : à partir du HTML, on souhaite récupérer les fichiers d'UE qui vous étaient fournis en entrée au TP3, au format suivant :

Numéro étudiant,Nom,Groupe>Note

Votre projet doit avoir la structure suivante :

- Un dossier principal, dans lequel se trouve vos scripts ;
- Un sous-dossier site contenu dans le dossier principal ;
- Un sous-dossier out également contenu dans le dossier principal, dans lequel seront placés les nouveaux fichiers produit par vos scripts.

À la fin de l'épreuve, placez tous vos scripts (et seulement vos scripts) dans une archive que vous mettrez en ligne sur Tomuss.

1 Parser la page web d'un étudiant

Dans cet exercice, on cherche à extraire tous les données utiles de la page web d'un-e étudiant-e.

Question 1. Écrivez un script qui prend en paramètre le nom d'un fichier HTML (page web de l'étudiant-e) et qui affiche le numéro étudiant. Si le nom de fichier fourni ne correspond pas à une page web d'étudiant-e, le script doit afficher un message d'erreur.

Aide : il est inutile de lire le contenu du fichier, le numéro d'étudiant est présent dans le nom du fichier.

Exemple d'utilisation :

```
$ ./get_numetu.sh site/etu_p2191623.html
p2191623
$ ./get_numetu.sh truc
Format de nom de fichier incorrect
$ ./get_numetu.sh site/ue_lifmachin.html
Format de nom de fichier incorrect
```

Question 2. Écrivez un script qui prend en paramètre le nom de la page web d'un-e étudiant-e, et qui affiche le nom de l'étudiant-e. Le script doit afficher une erreur si le fichier n'existe pas.

Exemple d'utilisation :

```
$ ./get_nom.sh site/etu_p2191623.html
Jeremy Irons
$ ./get_nom.sh truc
Fichier introuvable: truc
```

Question 3. Écrivez un script qui prend en paramètre le nom de la page web d'un-e étudiant-e, et qui affiche la liste de ses notes, au format présenté dans l'exemple ci-dessous.

Aide : c'est nettement plus simple avec `sed` qu'avec `grep/cut`.

Exemple d'utilisation :

```
$ ./get_notes.sh site/etu_p2191623.html
lifbidule 9
lifchose 7
lifmachin 7
liftruc 18
liftrucmuche 0
```

2 Générer les fichiers CSV

Dans cet exercice, on cherche à utiliser les données extraites des pages web pour générer le fichier CSV des notes d'une UE.

Question 1. Écrivez un script qui prend en argument le code d'une UE et un numéro étudiant, et qui affiche le groupe de cet-te étudiant-e dans cette UE.

Aide : la page web d'un groupe (par exemple `ue_liftruc_A.html`) contient les numéros des étudiant-es qui appartiennent au groupe. L'idée est de parcourir les pages web des groupes de l'UE pour trouver celui dans lequel se trouve le numéro étudiant recherché.

Exemple d'utilisation :

```
$ ./get_groupe.sh liftruc p2191623
D
```

Question 2. Écrivez un script qui prend en argument un code d'UE et le nom de la page web d'un-e étudiant-e, et qui affiche les informations de cet-te étudiant-e dans cette UE au format CSV (voir exemple). Si le nom de fichier est mal formaté ou n'existe pas, un message d'erreur doit s'afficher.

Aide : utilisez vos scripts des questions précédentes, ne les réécrivez pas. Vous pouvez aussi éviter de refaire l'affichage des messages d'erreur, en utilisant les codes de retour de vos scripts.

Exemple d'utilisation :

```
$ ./get_etu.sh liftruc site/etu_p2191623.html
p2191623,Jeremy Irons,D,18
$ ./get_etu.sh liftruc site/ue_liftruc_A.html
Format de nom de fichier incorrect
$ ./get_etu.sh liftruc site/etu_p9999999.html
Fichier introuvable: site/etu_p9999999.html
```

Question 3. Écrivez un script qui prend en paramètre un code d'UE, puis crée le répertoire `out` s'il n'existe pas déjà, et génère le fichier CSV de cette UE dans le dossier `out`.

Aide : il faut parcourir toutes les pages web des étudiant-es, et appeler le script de la question précédente sur chacun.

```
$ ./generer_ue.sh liftruc
$ cat out/liftruc | wc -l
137
$ cat out/liftruc | head -n 1
p0023892,Lionel Barrymore,C,19
$ cat out/liftruc | tail -n 1
p9935885,Broderick Crawford,B,4
```

Question 4. Écrivez un script qui génère les fichiers CSV de toutes les UE.

Exemple d'utilisation :

```
$ ./generer_toutes_ues.sh
$ ls out
lifbidule lifchose lifmachin liftruc liftrucmuche
```