

Système - TP3

Mini projet

Le but de ce TP est de réaliser un script (ou plutôt, un ensemble de scripts) plus complexe que ceux des TP précédents. Pour cela vous aurez besoin de mettre en application les notions vues aux TP 1 et 2.

En plus de ce sujet, vous trouverez un fichier d'archive **notes.tar.gz**. Cette archive contient des fichiers au format CSV. Chaque fichier représente les notes d'une UE imaginaire pour des étudiant-es (presque) imaginaires. Chaque ligne du fichier a la forme suivante :

Numéro étudiant,Nom de l'étudiant-e,Groupe de l'étudiant-e dans cette UE,Note obtenue

L'objectif est d'écrire un script qui, à partir de cette archive, est capable de générer automatiquement un site web simple en HTML présentant toutes les informations contenues dans les fichiers de notes. Il ne s'agit pas d'un TP de web : des connaissances minimales en HTML sont suffisantes, et il ne vous est pas demandé de faire de CSS ou de JavaScript.

Votre script devra également fonctionner avec une archive de notes différentes mais au même format. Ne partez donc pas du principe qu'une UE ou un-e étudiant-e précis-e sont forcément présents dans les fichiers. En revanche, vous pouvez présumer que tous les fichiers contiennent les mêmes étudiant-es (disons qu'il s'agit d'une formation en tronc commun, sans options). Un-e étudiant-e n'est pas forcément dans le même groupe dans chaque UE.

Vous êtes libres de piocher les outils et commandes que vous utiliserez parmi ceux vus aux TP précédents. Il est par exemple possible d'utiliser exclusivement du bash, ou exclusivement du awk, ou un mélange des deux.

Pour extraire l'archive, vous pouvez utiliser la commande suivante :

```
tar xzf notes.tar.gz
```

Cette commande extrait l'archive dans le dossier courant.

Votre programme utilisera l'arborescence de fichiers suivante :

- un dossier de travail principal, dans lequel vos scripts seront stockés ;
- un sous dossier **notes** dans lequel seront stockés les fichiers de notes ;
- un sous dossier **out** dans lequel seront stockés les fichiers HTML générés par vos scripts.

1 Scripts utiles

Dans cet exercice, on cherche à programmer de petits scripts utilitaires qui pourront être utilisés pour faciliter les exercices suivants.

Question 1. Écrivez un script qui, à partir d'une liste de valeurs entières, calcule et affiche la moyenne de ces valeurs.

Exemple d'utilisation :

```
$ ./moyenne 8 12 15  
11.6667
```

Attention, bash ne gère que les divisions entières. Par exemple :

```
$ echo $(( (8+12+15)/3 ))  
11
```

Si vous choisissez d'écrire ce script en bash, vous pourrez par exemple demander à awk de faire la division pour vous, comme suit :

```
$ awk "BEGIN {print (8+12+15)/3}"
11.6667
```

Ou bien vous pouvez choisir d'écrire le script entier en awk. Dans ce cas, il est plus simple de recevoir les notes sur l'entrée standard plutôt qu'en paramètre.

Question 2. Écrivez un script qui, pour une UE dont le code est donné en paramètre, affiche tous les groupes de cette UE.

Exemple d'utilisation :

```
$ ./lsgroupes lifbidule
A
B
```

Notez que la commande `uniq` est bien utile ici.

2 Page web d'un·e étudiant·e

Le but de cet exercice est de générer la page HTML présentant toutes les données d'un·e étudiant·e. Cette page web devra comporter les informations suivantes :

- Le nom et le numéro étudiant ;
- La moyenne de toutes les notes de l'étudiant·e (présumez que toutes les UEs ont un coefficient égal) ;
- La liste des notes de l'étudiant·e dans chaque UE.

Le fichier HTML sera généré dans le dossier `out`, et portera un nom de la forme `etu_pXXXXXX.html`

Voici par exemple le fichier `out/etu_p1537402.html` qui pourrait être généré par votre script :

```
<html><head><title>Jennifer Jones (p1537402)</title></head>
<body><h1>Jennifer Jones (p1537402)</h1>
<p>Moyenne: 9.6</p>
<table>
<tr><td>lifbidule</td><td>5</td></tr>
<tr><td>lifchose</td><td>14</td></tr>
<tr><td>lifmachin</td><td>3</td></tr>
<tr><td>liftruc</td><td>8</td></tr>
<tr><td>liftrucmuche</td><td>18</td></tr>
</table></body></html>
```

Question 1. Écrivez un script qui calcule la moyenne d'un·e étudiant·e en fonction de son numéro.

Exemple d'utilisation :

```
$ ./moyenne_etu p1537402
9.6
```

Pensez bien sûr à réutiliser le script de l'exercice précédent.

Question 2. Écrivez un script qui génère la page HTML d'un·e étudiant·e, comme décrit plus haut.

Exemple d'utilisation :

```
$ ./page_etu p1537402
$ ls out
etu_p1537402.html
```

Question 3. Écrivez un script qui génère toutes les pages HTML d'étudiant·es.

Exemple d'utilisation :

```
$ ./tous_etu
$ ls out | wc -l
137
```

3 Page web d'un groupe

On voudrait une page web qui liste les étudiant·es d'un groupe ainsi que leurs notes et leur moyenne. Rappelez vous que les groupes ne sont pas les mêmes dans chaque UE : il faut donc une page pour le groupe A de liftruc, une page pour le groupe A de lifbidule, etc.

Chaque page de groupe devra comporter les informations suivantes :

- Le code de l'UE et la lettre du groupe ;
- La moyenne des étudiant·es du groupe dans cette UE ;
- La liste des étudiant·es et leurs notes, avec un lien hypertexte vers la page de chaque étudiant·e.

Le fichier HTML produit devra être placé dans `out` et être nommé `ue_lifXXXX.Y.html`, où `lifXXXX` est le code de l'UE et `Y` est la lettre du groupe.

Question 1. Écrivez un script qui calcule la moyenne d'un groupe dans l'UE concernée.

Exemple d'utilisation :

```
$ ./moyenne_groupe liftruc A
11.9714
```

Question 2. Écrivez un script qui génère la page HTML d'un groupe donné.

Exemple d'utilisation :

```
$ ./page_groupe liftruc A
$ ls out | grep ue
ue_liftruc_A.html
```

Question 3. Écrivez un script qui génère les pages HTML de tous les groupes d'une UE donnée.

Exemple d'utilisation :

```
$ ./groupes_ue lifbidule
$ ls out | grep ue
ue_lifbidule_A.html
ue_lifbidule_B.html
```

N'oubliez pas que vous pouvez utiliser votre script de l'exercice 1 pour lister les groupes d'une UE.

4 Page web d'une UE

On voudrait une page web pour chaque UE. Chaque page devra contenir les informations suivantes :

- Le code de l'UE ;
- La moyenne des étudiant·es dans cette UE ;
- La liste des groupes de l'UE, avec un lien vers la page de chaque groupe.

Le fichier HTML produit sera placé dans `out` et portera un nom de la forme `ue_lifXXXX.html`.

Question 1. Écrivez un script qui calcule la moyenne des étudiant·es d'une UE.

Exemple d'utilisation :

```
$ ./moyenne_ue liftruc
11.1898
```

Question 2. Écrivez un script qui génère la page HTML d'une UE, ainsi que les pages de tous les groupes de cette UE.

Exemple d'utilisation :

```
$ ./page_ue lifbidule
$ ls out | grep lifbidule
ue_lifbidule_A.html
ue_lifbidule_B.html
ue_lifbidule.html
```

Question 3. Écrivez un script qui génère les pages web de toutes les UE.

5 Index du site

Finalement, il ne nous reste plus qu'à générer une page comportant la liste des UEs, avec un lien hypertexte vers la page de chaque UE.

Ce fichier HTML s'appellera `out/index.html`.

Question 1. Écrivez un script qui génère le fichier d'index.

Question 2. Écrivez un script qui crée les dossiers `notes` et `out`, extrait le contenu d'une archive fournie en paramètre dans le dossier `notes`, puis génère les pages web de tous les étudiant-es, de toutes les UEs, et l'index.

Exemple d'utilisation :

```
$ ls out
ls: impossible d'accéder à 'out': Aucun fichier ou dossier de ce type
$ ./generer_site notes.tar.gz
$ ls out | wc -l
159
```

Question 3. Ouvrez le fichier `index.html` dans votre navigateur web, et vérifiez que votre site fonctionne bien.