

L2 MIASHS – Informatique S3

TD4 : Chaînes de caractères

Rappels : les exercices fondamentaux doivent être compris et appris. Il y a leur correction sur l'EPI.

Je vous conseille aussi de travailler la correction des autres exercices, c'est une bonne manière de progresser.

Vous trouverez sur l'EPI deux fichiers `data.txt` et `data1.csv` qui sont utiles pour les exercices 4 et 6. Si Python vous rétorque que `No such file`, alors que vous avez téléchargé les fichiers, c'est sans doute qu'ils ne sont pas dans le répertoire où se trouve votre code Python.

1 Exercices fondamentaux

Exercice 1

Que fait le programme suivant ? Explicitez les affichages.

```
1 from re import findall
2
3 s = "Bonjour"
4 s += " tout le monde"
5 print(s[0]+s[-4:-6:-1]+"b"+ s[-1])
6 for x in s.split()[2]:
7     print(x)
8 p = "{} patates coutent {} euros."
9 print(p.format(10,2))
10 print(s.upper())
11 print(s.isnumeric())
12 # s[3] = 'a'
```

Exercice 2

Proposez une fonction qui prend en paramètre une chaîne de caractères et retourne la chaîne obtenue en ajoutant une majuscule au début de chaque mot.

Par exemple, pour la chaîne `"l'objectif est de trouver dans un texte non"`, la fonction retourne : `"L'objectif Est De Trouver Dans Un Texte Non"`.

Indication : On pourra utiliser, `split`, `join` et `capitalize`.

Remarque : On ne peut pas directement affecter une valeur à un caractère de la chaîne. Rappelons-nous que les `str` sont immuables.

Exercice 3

Proposez une fonction qui double tous les nombres entiers isolés présents dans une chaîne de caractère.

Exemple "3 kg 12 euros. 0.25 melon, 1 euros. Par contre mon3. donnera :
"6 kg 24 euros. 0.25 melon, 2 euros. Par contre mon3.

Indication : On pourra utiliser `isnumeric` pour déterminer si une chaîne de caractères représente un nombre, et s'inspirer de l'exercice précédent.

Exercice 4

L'objectif est d'écrire une fonction qui ouvre un fichier dont le nom est `data.txt` et construit un fichier `data5.txt` qui contienne tous les mots de longueur 5 du fichier précédent. Pour cela on vous demande d'écrire les fonctions suivantes :

1. Une fonction `charge(nom)` qui prend un chemin d'accès à un fichier en paramètre (ou un nom de fichier), et retourne une liste de chaînes de caractères qui sont les mots contenus dans le fichier.
2. Une fonction `selecteur_longueur(1,n)` qui prend en paramètre une liste de chaînes de caractères `1` et un entier `n` et retourne une liste constituée des chaînes de `1` de longueur `n`.
3. Une fonction `sauve(nom, 1)` qui écrit dans le fichier `nom` les chaînes de caractères de la liste `1` séparées par des espaces.
4. une fonction qui répond à la question initiale en utilisant les fonctions précédentes.

Remarque Quand on parle de *mot*, ce n'est pas au sens de la langue française mais au sens de morceaux de chaînes de caractères entourées par des espaces. Par exemple "z43a," peut être un mot.

2 Exercices complémentaires

Exercice 5

Proposez une fonction qui à partir d'une longue chaîne `s` (>100) renvoie une liste de chaînes de caractères contenant chacune 24 caractères maximum, sans découper les mots. Chaque chaînes se termine donc par un espace. Et si un mot ne rentre pas, plutôt que le découper, on le rajoute à la chaîne suivante.

Affichez ces chaînes ligne par ligne.

Exercice 6

Le fichier `data1.csv` contient un extrait de la célèbre base de données Titanic. On a conservé les enregistrements pour lesquels les données sont complètes pour les champs `pclass` (qui correspond à la classe dans laquelle la personne voyage), `survived`, `name`, `sex`, `age` et `fare`.

1. Chargez ces données et affichez le début de manière à comprendre comment sont stockées les données dans ce fichier csv.

2. Rangez les données dans une liste de dictionnaires, chaque dictionnaire correspondra à un enregistrement, avec comme clés, les noms des colonnes et comme valeur associée à une clé, la valeur pour l'enregistrement.
3. Comptez combien d'homme et de femmes ont survécu, affichez le résultat sous forme de tableau formatée proprement.
4. Calculez le taux de survivant selon des classes d'âges et le sexe puis affichez le résultat de manière formatée proprement dans un tableau.
5. Cherchez du côté de `matplotlib`, pour afficher des diagrammes qui illustrent les résultats précédents