
TD 1

PYTHON : LES BASES ET UN PEU PLUS

Exercice 1. Les types et les conteneurs

Ouvrir python en ligne de commande.

- Vérifier les types des valeurs suivantes: (si certains de ces types sont encore inconnus, relire le cours sur les bases en python !)
`4; 4.0; -1; 2.4; None; a; 'hello'; "hello"; 0; True; False`
`[1, 2]; [1, 2.4]; ['a', 'b']; (1,3,5); ("a", 2); {'one':'un', 'zero':'zéro'}`
- Affecter les variables suivantes: `var_int = 1; var_float = 2.51; var_str = 'hello'; var_bool = True`
Convertir les types de ces quatre variables en utilisant les fonctions `int()`; `float()`; `str()`; `bool()`.
Vérifier les valeurs et les types obtenus après conversion.
En particulier remarquer le résultat de `bool(var_int)` et `int(var_float)`.
Vous pouvez bien sûr tester d'autres combinaisons, par curiosité.
- Pour les variables `var`, donner les valeurs qu'elles contiennent (on pourra utiliser la commande `print(var)`), leur type. Faites d'autres propositions permettant d'obtenir le même contenu.
Que retourne `var[2]` ? Quel effet a la commande `var[2] = 6`, pourquoi ? Si il y a une erreur, proposer une commande adaptée qui ne génère pas d'erreur.
 - `var = list(range(10))`
 - `var = "informatique"`
 - `var = {"info":3, "maths":5, "musique":10, "foot":7}`

Exercice 2. Les entrées / sorties

- Avec la fonction `input()` créer une variable `var` contenant une valeur qui aura été saisie à la main au clavier. Afficher cette variable avec la fonction `print`.
- Créer un fichier (avec `vim` par exemple) contenant le texte suivant:
`MAN: Mise à niveau`
`INFO: Master d'informatique`
Ouvrir ce fichier avec la commande `f = open("nomFichier","modeAccès",encoding="utf8")` et les arguments appropriés. Afficher les lignes les unes après les autres (boucle `for` puis `print`).
Ecrire une troisième ligne avec la fonction `write()` contenant `M1: première année de master` au fichier créé initialement.
Attention: ne pas oublier de REFERMER le fichier ouvert.

Exercice 3. Écrire un programme en python

Vous avez à disposition un dictionnaire de prononciations phonétiques. Une prononciation est une séquence de symboles phonétiques séparés par des espaces. Pour chaque ligne, vous disposez d'un mot, de sa prononciation phonétique donnée selon l'alphabet X-Sampa et de sa prononciation phonétique donnée selon l'alphabet IPA. Vous ne considérerez que la prononciation donnée en X-Sampa. Les mots et les prononciations sont séparées par des point-virgules.

Préliminaire:

- Dans un répertoire, créer un fichier et le nommer `hello_world.py`.
 - Ouvrir le fichier `hello_world.py` avec votre éditeur préféré (`vim`, `notepad`, `kate`, `textedit`, etc...) puis y inscrire le code suivant:

```
ma_chaine="Hello World" print(ma_chaine)
```
 - Lancer votre programme dans un terminal à partir du bon répertoire avec la commande:

```
python hello_world.py
```
 - Si tout va bien, vous verrez s'afficher à l'écran la chaîne de caractères.
1. Créer un programme python qui lit le document `synpaflex-pronunciation-dictionary.txt` et affiche les n premières lignes (équivalent à la commande unix `head`)
 2. Pourquoi y a-t-il une ligne vide entre chaque affichage? Corriger cela avec la fonction `.strip()`
 3. Maintenant, créer une fonction `main()` qui ouvre le fichier, puis une fonction `AfficherLigne(line)` qui prend en argument une ligne et l'affiche à l'écran.
 4. Créer une nouvelle fonction `StockerLigne(line, dicPron)` qui stocke le contenu d'une ligne dans un dictionnaire (initialisation dans la fonction `main()` et ajout d'éléments dans la fonction) et retourne le dictionnaire modifié. Les clés seront les mots. Les valeurs seront des prononciations suivant l'alphabet IPA. Penser à supprimer les espaces en début et fin de chaîne de caractère.
 5. Créer une fonction `AfficherPron(m, dicPron)` qui affiche à l'écran la prononciation d'un mot. Donner la prononciation du mot `coulé`.
Rq: pour gérer les accents en Français, il faudra ajouter en entête du programme la ligne suivante:

```
#-*- coding: utf-8 -*-
```
 6. Créer une fonction `AfficherMots(p, dicPron)` qui stocke l'ensemble des mots qui ont une prononciation donnée dans une liste, les compte et les affiche. Donner les mots qui se prononcent `/k u l e/` et les compter.