

TPinfo : Utilisation de dictionnaires

Objectifs pédagogiques du TP :

- Découverte du type dictionnaire.
- Comptage des éléments d'un tableau à l'aide d'un dictionnaire.

I Dictionnaires en langage Python : découverte

On a vu que les éléments d'une liste sont ordonnés et on accède à un élément grâce à sa position dans la liste en utilisant ce qu'on appelle l'**indice** de l'élément. Un **dictionnaire** va aussi permettre de rassembler des éléments mais ceux-ci seront identifiés par une **clé**.

Exemple : Dans un dictionnaire papier, on accède à une définition (l'élément) à partir d'un mot (la clé).

1. Ouvrir le fichier decouverte_dictionnaire.py et executer successivement les différentes lignes afin de manipuler des données du type dictionnaire

Exemple ou exercice d'application – Définir un dictionnaire

```
dico = { } .....
dico[clé] = valeur .....
dico = {clé1 : valeur1 , clé2 : valeur2 , clé3 : valeur3} .....
dico = dict([[clé1,valeur1],[clé2,valeur2],[clé3,valeur3]]) .....
dico = {x : x**2 for x in range(10)} .....
```

Remarques :

- Les dictionnaires sont des objets mutables : on peut ajouter, modifier, supprimer le contenu.
- L'ordre des éléments dans un dictionnaire n'a pas d'importance.
- Les dictionnaires sont de type dict.

Quelques méthodes et fonctions sur les dictionnaires

Exemple ou exercice d'application – Afficher, modifier, supprimer une entrée d'un dictionnaire

```
len(dico) .....
dico[clé] .....
dico[clé]=nouvelleValeur .....
clé in dico .....
del dico[clé] .....
dico.clear() .....
```

Exemple ou exercice d'application – Accéder à la liste des clés ou des valeurs

```
list(dico.keys()) .....
list(dico.values()) .....
list(dico.items()) .....
```

Remarque : L'ordre renvoyé des listes des clés, valeurs ou couples (clé,valeur) est aléatoire !


Copie d'un dictionnaire

Comme pour les listes, l'affectation d'un dictionnaire (à partir d'un dictionnaire préexistant) crée en réalité une **référence** et non une **copie**.

 **Exemple ou exercice d'application** – Alias (référence) ≠ copie

```
dico1 = dico .....
dico2 = dict(dico) .....
```

Parcourir un dictionnaire

 **Exemple ou exercice d'application** – Itération sur les clés ou sur les valeurs

```
for clé in dico .....
for clé in dico.keys() .....
for valeur in dico.values() .....
for clé,valeur in dico.items() .....
```

“Naturellement”, la variable d’itération d’une boucle `for` sur un dictionnaire est une clé. S’il est nécessaire d’itérer plutôt sur les valeurs, on pourra utiliser la méthode `.values()`. S’il est nécessaire d’itérer plutôt sur les couples (clé,valeur), on pourra utiliser la méthode `.items()`.

II Mots les plus fréquents dans un texte

Le script fournit contient une variable `texte` contenant une chaîne de caractère (ici une courte biographie d’Ada Lovelace). On écrira les fonctions de cette partie à la suite de ce script de manière à pouvoir utiliser la variable `texte` si besoin.

2. Écrire une fonction `occurrences(lst)` qui prend en argument une liste `lst` et qui renvoie un dictionnaire `dic` donnant, pour chaque valeur apparaissant dans `lst`, le nombre de fois qu’elle apparaît dans `lst`.
Indice : pour attribuer la valeur `val` à la clé `x` dans le dictionnaire `dic`, il faut exécuter la commande `dic(x) = val`.
3. Tester la fonction `occurrences` sur la liste `lst=[1,2,3,2,1,4,1,2,1,2,3]`.
4. Écrire une fonction `plusFrequents(lst)` qui prend en argument une liste `lst` et qui renvoie la liste de la ou des valeurs ayant le plus grand nombre d’occurrences dans la liste `lst` ainsi que le nombre d’apparitions correspondant.
5. Tester la fonction `plusFrequents` sur la liste `l=[1,2,3,2,1,4,1,2,1,2,3]`.
6. Taper dans le Shell

```
>>> texte
>>> lst=texte.split()
>>> lst
>>> lst.count('.')
```

7. Écrire une fonction `bioSansPonctuation(texte)` qui renvoie une liste contenant tous les mots de la chaîne de caractère `texte` en enlevant les caractères `'.'` et `','`.
8. En utilisant les fonctions précédentes, déterminer le ou les mots ayant le plus grand nombre d’occurrences dans la biographie fournie d’Ada LOVELACE.

III Jouer gagnant au Scrabble

Le script fournit définit un dictionnaire dont les clés sont les 26 lettres de l'alphabet et les valeurs sont les points bruts rapportés par chacune de ces lettres au Scrabble. On écrira les fonctions de cette partie à la suite de ce script de manière à pouvoir utiliser le dictionnaire `scrabble`.

Le fichier `quelquesMotsAlea.txt` est un fichier contenant 986 mots français choisis aléatoirement parmi plus de 330000 mots du dictionnaire (sans espace, apostrophe, tiret ou accent).

9. écrire une fonction `points` qui prend en argument une chaîne de caractère correspondant à un mot (sans espace, apostrophe, tiret ou accent) et qui renvoie le nombre de points minimum que rapporte ce mot au Scrabble.

⚠ *Attention* : dans le dictionnaire `scrabble`, les clés sont données en majuscules. On prendra soin que la fonction renvoie le bon résultat y compris si le mot fourni en argument est écrit en minuscule. On pourra utiliser la méthode `.upper()` : `'blabla'.upper()` renvoie `'BLABLA'`.

10. Tester la fonction `points` pour déterminer combien de points bruts rapporte le mot « tortue ».
11. Écrire une fonction `extraction` sans argument qui lit le fichier `quelquesMotsAlea.txt` et renvoie la liste des mots qu'il contient.
12. Écrire une fonction `maxScore` qui prend en argument une liste de mots `lst` et qui renvoie le ou les mots de la liste `lst` permettant de réaliser le meilleur score au Scrabble, ainsi que le score correspondant.
13. Tester la fonction `maxScore` sur la liste des mots du fichier `quelquesMotsAlea.txt`.
14. Si vous avez déjà joué au Scrabble, vous savez qu'on ne peut en fait proposer que des mots de 7 lettres maximum... Écrire une fonction `maxScoreVrai` qui prend en argument une liste de mots `lst` et qui renvoie le ou les mots **jouables** de la liste `lst` permettant de réaliser le meilleur score au Scrabble, ainsi que le score correspondant.
15. Tester la fonction `maxScorePourDeVrai` sur la liste des mots du fichier `quelquesMotsAlea.txt`.

IV Traduction d'un texte en morse

Le script proposé définit un dictionnaire dont les clés sont les 26 lettres de l'alphabet et les valeurs correspondantes sont le code morse associé à chaque lettre. On écrira les fonctions de cette partie à la suite de ce script de manière à pouvoir utiliser le dictionnaire `morse` et les variables `messageTest` et `codeTest` si besoin.

16. Écrire une fonction `tradMorse` qui prend en argument un message (une chaîne de caractère donnée en majuscules, sans accents, sans ponctuation) et qui traduit ce texte en morse avec 3 espaces entre chaque lettre et 7 espaces entre chaque mot.
17. Tester la fonction `tradMorse` sur le message `'LA VIE EST BELLE'` (c'est la variable `messageTest` définie en début de script).
18. Écrire une fonction `dico_morseFr` sans argument qui retourne un dictionnaire dont les clés sont les caractères morses et les valeurs les lettres associées.
19. Écrire une fonction `tradFr` qui traduit un code en morse un texte en français.
20. Tester la fonction `tradFr` sur le code en morse donné dans la variable `codeTest` définie en début de script.