

FICHE : DICTIONNAIRES EN LANGAGE PYTHON

DICTIONNAIRES EN LANGAGE PYTHON

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).

Définir un dictionnaire

```
dico = {} ..... dictionnaire vide

dico[clé] = valeur ..... ajout d'entrées au fur et à mesure

dico = {clé1 : valeur1 , clé2 : valeur2 , clé3 : valeur3} ..... ajout d'entrées en une seule fois

dico = dict([[clé1,valeur1],[clé2,valeur2],[clé3,valeur3]]) ..Création d'un dictionnaire
en convertissant une liste de liste à 2 éléments

dico = {x : x**2 for x in range(10)} ..... définition d'un dictionnaire par compréhension
```

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

Afficher, modifier, supprimer une entrée d'un dictionnaire

```
len(dico) ..... nombre d'entrées

dico[clé] ..... donne la valeur associée à la clé

dico[clé]=nouvelleValeur .....
ajoute la valeur associée à la clé ou modifie la valeur associée à la clé si elle existait déjà

clé in dico ..... teste si le dictionnaire possède une entrée correspondant à la clé

del dico[clé] ..... supprime l'entrée du dictionnaire correspondant à la clé

dico.clear() ..... vide le dictionnaire
```

Accéder à la liste des clés ou des valeurs

```
list(dico.keys()) ..... donne la liste de toutes les clés du dictionnaire  
list(dico.values()) ..... donne la liste de toutes les valeurs du dictionnaire  
list(dico.items()) ..... donne la liste de tous les couples (clé,valeur) du dictionnaire
```

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

COPIE DE 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**.

Alias versus copie

```
dico1 = dico crée un alias (une référence) du dictionnaire dico - si dico est modifié, dico1 aussi et vice-versa  
dico2 = dict(dico) ..... crée une copie indépendante du dictionnaire dico
```

PARCOURIR UN DICTIONNAIRE

« Naturellement », la variable d'itération d'une boucle `for` sur un dictionnaire est une clé.

```
for clé in dico :
```

ce qui est équivalent à

```
for clé in dico.keys() :
```

S'il est nécessaire d'itérer plutôt sur les valeurs, on pourra utiliser la méthode `.values()`.

```
for valeur in dico.values() :
```

S'il est nécessaire d'itérer plutôt sur les couples (clé,valeur), on pourra utiliser la méthode `.items()`.

```
for clé,valeur in dico.items() :
```