



CHAPITRE 8 : LES DICTIONNAIRES

Comme dans un dictionnaire ordinaire où les éléments sont des couples (mot : explication), les éléments des **dictionnaires Python** sont des couples (**clé : valeur**) :

- **Les clés** : sont les index, qui peuvent être de n'importe quel type non-modifiable et chaque clé est unique dans un même dictionnaire (les chaînes et les nombres peuvent toujours être des clés).
- **Les valeurs** : peuvent être de n'importe quel type (int, float, str, list, tuple ..)

On construit explicitement un dictionnaire par une expression de la forme :

{ clé1 : valeur1 , clé2 : valeur2 , ... clék : valeurk }

Si **d** est un dictionnaire, les opérations supportées par **d** sont les suivantes :

- **len(d)** : retourne le nombre d'éléments de d ;
- **d[i]** : retourne la valeur associée à la clé i .

I. Définir un dictionnaire:

<pre>a = {} print (a)</pre>	#définir un dictionnaire vide	le script affiche : { }
<pre>a = dict() print (a)</pre>	#définir un dictionnaire vide	le script affiche : { }
<pre>a = {'point': 1, 'ligne': 2, 'triangle': 3} print (a)</pre>		le script affiche : {'ligne': 2, 'triangle': 3, 'point': 1}

Remarques :

- Un dictionnaire **n'est pas** une **séquence** (il n'y a pas d'ordre sur les éléments), mais est un objet modifiable (comme les listes).
- Les dictionnaires ne sont pas des séquences
- On **ne peut pas concaténer** un dictionnaire avec un autre

II. Accès à un élément individuel : dict[clé]

Exemple :

<pre>d= {'MPSI' :3,'PCSI' :1, 1: 'ECT' } print(d['MPSI']) print(d[1])</pre>	le script affiche : 3 ECT
---	---------------------------------

**III. Test d'appartenance (sur les clés) : in****Exemple :**

```
d= {'MPSI' :3,'PCSI' :1, 1: 'ECT' }
b='PCSI' in d
print(b)
b='MP' in d
print(b)
```

le script affiche :
True
False

IV. Ajout d'une clé, modification d'une valeur, la suppression d'un élément :

```
a = {'point': 1, 'ligne': 2, 'triangle': 3}
print(a)
a['cercle'] = 4           #Ajout d'une clé
print(a)
a['point'] = 3           #Modifier une valeur
print(a)
del(a['triangle'])      # la suppression de couples
('triangle': 3)
print(a)
```

le script affiche :
{'ligne': 2, 'point': 1, 'triangle': 3}
{'ligne': 2, 'point': 1, 'cercle': 4, 'triangle': 3}
{'ligne': 2, 'point': 3, 'cercle': 4, 'triangle': 3}
{'ligne': 2, 'point': 3, 'cercle': 4}

V. Parcourir les dictionnaires.**Afficher les clés :**

```
a = {'point': 1, 'ligne': 2, 'triangle': 3}
for k in a:
    print(k)
```

le script affiche :
point
ligne
triangle

Afficher les valeurs :

```
a = {'point': 1, 'ligne': 2, 'triangle': 3}
for k in a:
    print(a[k])
```

le script affiche :
2
3
1

Afficher les tuples (clé,valeur) :

```
a = {'point': 1, 'ligne': 2, 'triangle': 3}
for k in a.items():
    print(k)
```

le script affiche :
('ligne', 2)
('point', 1)
('triangle', 3)

Afficher les clés et les valeurs :

```
a = {'point': 1, 'ligne': 2, 'triangle': 3}
for i,j in a.items():
    print(i,j)
```

le script affiche :
triangle 3
ligne 2
point 1

**VI. Attention !! Copie d'un dictionnaire :**

<pre>a= {'age':18, 'prénom':'karim'} b= a b['classe'] = 'MPSI' print(b) print(a)</pre>	le script affiche : <pre>{'prénom': 'karim', 'age': 18, 'classe': 'MPSI'} {'prénom': 'karim', 'age': 18, 'classe': 'MPSI'}</pre>
<pre>a= {'age':18, 'prénom':'karim'} c = a.copy() c['classe'] = 'MPSI' print(c) print(a)</pre>	le script affiche : <pre>{'prénom': 'karim', 'age': 18, 'classe': 'MPSI'} {'prénom': 'karim', 'age': 18}</pre>

VII. Un dictionnaire a ses propres méthodes :(help(dict))

<ul style="list-style-type: none"> • keys () : retourne une liste de toutes les clés utilisées dans le dictionnaire, dans un ordre quelconque 	<pre>a={'point': 1, 'ligne': 2, 'triangle': 3} L=a.keys() print(L)</pre>	le script affiche : <pre>dict_keys(['ligne', 'point', 'triangle'])</pre>
<ul style="list-style-type: none"> • values () : renvoie une liste des valeurs 	<pre>a= {'point': 1, 'ligne': 2, 'triangle': 3} L=a.values() print(L)</pre>	le script affiche : <pre>dict_values([2, 3, 1])</pre>
<ul style="list-style-type: none"> • items () : retourne la liste des couples (clé, valeur) 	<pre>a= {'point': 1, 'ligne': 2, 'triangle': 3} L=a.items() print(L)</pre>	le script affiche : <pre>dict_items([('triangle', 3), ('ligne', 2), ('point', 1)])</pre>
<ul style="list-style-type: none"> • get() : comme dico[cle] 	<pre>a= {'point': 1, 'ligne': 2, 'triangle': 3} print(a.get('ligne'))</pre>	le script affiche : <pre>2</pre>
<ul style="list-style-type: none"> • clear() : supprime tous les couples du dictionnaire 	<pre>a= {'point': 1, 'ligne': 2, 'triangle': 3} a.clear() print(a)</pre>	le script affiche : <pre>{}</pre>
<ul style="list-style-type: none"> • pop(cle) : retire la clé et renvoie sa valeur 	<pre>a= {'point': 1, 'ligne': 2, 'triangle': 3} b=a.pop('ligne') print(a) print(b)</pre>	le script affiche : <pre>{'triangle': 3, 'point': 1} 2</pre>
<ul style="list-style-type: none"> • popitem() : retire et renvoie un couple clé-valeur 	<pre>a= {'point': 1, 'ligne': 2, 'triangle': 3} print(a) b=a.popitem() print(a) print(b)</pre>	le script affiche : <pre>{'ligne': 2, 'triangle': 3, 'point': 1} {'triangle': 3, 'point': 1} ('ligne', 2)</pre>