

Initiation à Python

Premiers pas

Dictionnaire

LIST

Laboratoire d'Informatique, Système et Télécommunications

Cours du Pr. Abdellah AZMANI
Département Génie Informatique
Cours pour les étudiants des Masters
et des Cycles d'Ingénieurs



Université Abdelmalek Essaadi
Faculté des Sciences et Techniques
Tanger



PYTHON

Premiers pas

Dictionnaire

Qu'est ce qu'un dictionnaire ?

Définition et Syntaxe

Python propose le type « dictionnaire » qui s'utilise comme une liste avec une variante puisqu'il s'appuie sur des clés ou index numérique, alpha ou alphanumérique.

un dictionnaire gère n'importe quel type : valeurs numériques, chaînes, listes, tuples, dictionnaires, et également des fonctions, des *classes* ou des *instances*

Pour initialiser une variable type dictionnaire, on procède ainsi :

```
>>> Dico = {}
```

Si nous voulons ajouter une entrée, nous allons indiquer une clé, ainsi que la valeur de cet élément :

```
>>> Dico = {}
```

```
>>> Dico['Cle_A'] = 'Analyse'
```

```
>>> Dico['Cle_B'] = 'Booléan'
```

```
>>> Dico
```

```
{'Cle_A': 'Analyse', 'Cle_B': 'Booléan'}
```

On peut observer qu'un dictionnaire correspond à une série enfermée entre deux accolades et dont les éléments sont séparés par des « , ». Chaque élément est formé d'un **index** ou **clé** et d'une **valeur**, séparés par un « : », on parle de **clé-valeur**.

PYTHON

Premiers pas

Dictionnaire

Méthodes et
fonctionnalités?

Affichage des valeurs d'un dictionnaire :

```
>>> Agenda = {"nom": "ASWAD", "NumFone": "0611223344"}
```

Méthode « keys » pour afficher les clés d'un dictionnaire :

```
>>> for K in Agenda.keys():
```

```
...     print (K)
```

```
...
```

```
nom
```

```
NumFone
```

Méthode « values » pour afficher toutes les valeurs d'un dictionnaire :

```
>>> for V in Agenda.values():
```

```
...     print (V)
```

```
...
```

```
ASWAD
```

```
0611223344
```

PYTHON

Premiers pas

Dictionnaire

Méthodes et
fonctionnalités?

Définition et Syntaxe

```
>>> Agenda = {"nom": "ABYAD ", "NumFone": "0644332211"}
```

Méthode « get » pour l’affichage d’une valeur dans un dictionnaire

```
>>> Agenda.get("nom")
```

ABYAD

Méthode « items » pour afficher les clés et les valeurs :

```
>>> for C,V in Agenda.items():
```

```
...     print (C, V)
```

```
...
```

nom ABYAD

NumFone 0644332211

PYTHON

Premiers pas

Dictionnaire

Méthodes et
fonctionnalités?

Définition et Syntaxe

```
>>> Agenda = {"nom" : "AZRAK", "NumFone": "0699887766"}
```

La méthode « copy » pour créer une copie d'un dictionnaire :

```
>>> CopyAgenda = Agenda.copy()
```

```
>>> Agenda ["nom"] = "AKHDAR"
```

```
>>> Agenda
```

```
{'NumFone': '0699887766', 'nom': 'AKHDAR'}
```

```
>>> CopyAgenda
```

```
{'NumFone': '0699887766', 'nom': 'AZRAK'}
```

Attention :

Si nous affectons, les 2 copies ne seront pas indépendantes :

```
>>> Agenda = {"nom": "WARDI", "NumFone": "0655667788"}
```

```
>>> CopyAgenda = Agenda
```

```
>>> Agenda ["nom"] = "BORTOKALI"
```

```
>>> Agenda
```

```
{"nom": "BORTOKALI", "NumFone": " 0655667788 "}
```

```
>>> CopyAgenda
```

```
{"nom": "BORTOKALI", "NumFone": " 0655667788 "}
```

PYTHON

Premiers pas

Dictionnaire

Méthodes et
fonctionnalités?

Exemple

Contrairement aux listes, il n'est pas nécessaire de faire appel à une méthode particulière (telle que **append()**) **pour ajouter de nouveaux éléments à un dictionnaire** : il suffit de créer une nouvelle paire **clé-valeur**.

```
>>> traducteur = {}  
>>> traducteur['intelligent'] = 'smart'  
>>> traducteur['ville'] = 'city'  
>>> traducteur['Tanger'] = 'Tangier'  
>>> print(traducteur)  
{'intelligent': 'smart', 'ville': 'city', 'Tanger': 'Tangier'}
```

Pour accéder à un élément, nous utilisons les clés :

```
>>> print(traducteur['ville'])  
city
```

PYTHON

Premiers pas

Dictionnaire

Méthodes et fonctionnalités?

La méthode **keys()** renvoie la sequence des *clés utilisées dans le dictionnaire*.

Cette sequence peut etre

utilisée telle quelle dans les expressions, ou convertie en liste ou en tuple si nécessaire, avec les fonctions integrees correspondantes : **list()** et **tuple()** :

```
>>> print(traducteur.keys())  
dict_keys(['intelligent', 'ville', 'Tanger '])
```

```
>>> for k in traducteur.keys():  
... print("cle :", k, " --- valeur :", traducteur[k])  
...
```

```
cle : intelligent --- valeur : smart  
cle : ville --- valeur : city  
cle : Tanger --- valeur : Tangier
```

```
>>> list(traducteur .keys())  
['computer', 'mouse', 'keyboard']
```

```
>>> tuple(traducteur .keys())  
('computer', 'mouse', 'keyboard')
```

PYTHON

Premiers pas

Dictionnaire

Méthodes et
fonctionnalités

#Affichage des clés

```
>>> for clef in traducteur:  
... print(clef)  
...
```

#Affichage des valeurs

```
>>> for clef in traducteur:  
... print(clef, traducteur[clef])  
...
```

#Affichage des valeurs version 2

```
>>> for clef, val in traducteur.items():  
... print(clef, val)  
...
```


PYTHON

Premiers pas

Dictionnaire TD

Exercices TD

Créer un dictionnaire Stock_FL pour gérer le stock des fruits et légumes suivants : Pomme, Banane, Orange, Raisin, Carote, Corgette, Petit Poids, Aubergine ...

Donner une solution pas à pas pour :

1. Créer Stock_FL
2. Ajouter les éléments
3. Afficher Stock_FL par interrogation ou vérification
4. Afficher Stock_FL par print
5. Afficher tout en parcourant Stock_FL (clé, valeur et clé-valeur)
6. Donner le nombre de catégorie de produits présents
7. Donner le nombre de produits présents
8. Copier Stock_FL dans Stock (cas de Copy et cas d'une affectation)
9. Que donnera l'instruction Stock_FL.items()
10. Supprimer un produit (avec del())
11. Vérifier les valeur de stock et Stock_FL. Que constatez vous ?
12. Utiliser le conditionnel pour afficher un message selon qu'un produit existe ou n'existe pas dans le stock (avec in ...)
13. Former une liste des produits présents en stock.
14. Former une liste avec les valeurs des quantités du stock
15. Former la liste et le tuple des produits et leur quantités disponible (avec stock.items()).

PYTHON

Premiers pas

Dictionnaire TD

Exercices TD suite

Créer une base de données fonctionnant à l'aide d'un dictionnaire, dans lequel on enregistre les noms des étudiants, leur moyenne et leur classement.

On suppose que le dictionnaire ne contient pas plusieurs valeurs identiques.