

Révisions dictionnaires

Rappels de cours

Avec Python, un dictionnaire est un type de données composé (contient plusieurs valeurs), un peu comme les listes.

- Les listes contiennent des valeurs « stockées » à des positions qui correspondent aux premiers nombres entiers de 0 à len(liste)-1.
- Les dictionnaires contiennent des valeurs « stockées » en les désignant par presque tout type de valeur, mais nous utiliserons le plus souvent des chaînes de caractères.

La syntaxe pour manipuler les dictionnaires est la même que pour les listes : `liste[k]` désigne la valeur stockée à la clé k.

Dans le dictionnaire de l'exemple, "gervais" est stocké à la clé "n".

```
>>> d = {"p": "ricky", "n": "gervais", "a": 63}
>>> d["p"]
'ricky'
>>> d["n"] = "earth"
>>> d
{"p": "ricky", "n": "earth", "a": 63}
>>> d["n"]
'earth'
>>> d.items() # liste les paires clé-valeur de d
[('p', 'ricky'), ('n', 'earth'), ('a', 63)]
>>> # pratique pour parcourir le dictionnaire !
```

Exercices

1. Écrire une fonction qui crée un dictionnaire ayant 26 clés : les 26 lettres de l'alphabet (majuscules), la valeur zéro étant attribuée à toutes les clés.
2. Écrire une fonction qui parcourt une chaîne de caractère et retourne un dictionnaire comme le précédent mais qui indique le nombre de lettres rencontrées dans la chaîne de caractères parcourue (combien de A, combien de B...
Testez avec un article de Wikipedia, par exemple celui sur « la fréquence d'apparition des lettres ».
3. Écrire une fonction qui donne la lettre la plus fréquente dans une chaîne de caractères.
4. On veut créer un carnet d'adresses, écrire une fonction qui interroge l'utilisateur et produit un dictionnaire décrivant un contact (nom, prénom, adresse...)

Révisions dictionnaires

Rappels de cours

Avec Python, un dictionnaire est un type de données composé (contient plusieurs valeurs), un peu comme les listes.

- Les listes contiennent des valeurs « stockées » à des positions qui correspondent aux premiers nombres entiers de 0 à len(liste)-1.
- Les dictionnaires contiennent des valeurs « stockées » en les désignant par presque tout type de valeur, mais nous utiliserons le plus souvent des chaînes de caractères.

La syntaxe pour manipuler les dictionnaires est la même que pour les listes : `liste[k]` désigne la valeur stockée à la clé k.

Dans le dictionnaire de l'exemple, "gervais" est stocké à la clé "n".

```
>>> d = {"p": "ricky", "n": "gervais", "a": 63}
>>> d["p"]
'ricky'
>>> d["n"] = "earth"
>>> d
{"p": "ricky", "n": "earth", "a": 63}
>>> d["n"]
'earth'
>>> d.items() # liste les paires clé-valeur de d
[('p', 'ricky'), ('n', 'earth'), ('a', 63)]
>>> # pratique pour parcourir le dictionnaire !
```

Exercices

1. Écrire une fonction qui crée un dictionnaire ayant 26 clés : les 26 lettres de l'alphabet (majuscules), la valeur zéro étant attribuée à toutes les clés.
2. Écrire une fonction qui parcourt une chaîne de caractère et retourne un dictionnaire comme le précédent mais qui indique le nombre de lettres rencontrées dans la chaîne de caractères parcourue (combien de A, combien de B...
Testez avec un article de Wikipedia, par exemple celui sur « la fréquence d'apparition des lettres ».
3. Écrire une fonction qui donne la lettre la plus fréquente dans une chaîne de caractères.
4. On veut créer un carnet d'adresses, écrire une fonction qui interroge l'utilisateur et produit un dictionnaire décrivant un contact (nom, prénom, adresse...)