

Les dictionnaires de Python

En cas de difficultés, pensez à relire la leçon avant de demander de l'aide au professeur.

I. EXERCICES D'ÉCHAUFFEMENT

. EXERCICE 1 : ACCÈS OU MODIFICATION

1. Écrire l'algorithme correspondant à ce code.

```
dico = {}
dico["house"]="maison"
dico["kitchen"]="cuisine"
dico["bed room"]="chambre"
print(dico)
print( dico["kitchen"] )
```

.....

2. Recopier et exécuter le code ci-dessus. Quel affichage obtient-on ?

.....

3. Ajouter à ce dictionnaire la clé « bunk beds » en lui associant la valeur « lits superposés ».
4. Afficher le dictionnaire pour vérifier que ce nouveau couple est bien présent.
5. Afficher la longueur de ce dictionnaire. Vérifier qu'elle est bien de 4.
6. Quelle instruction permet d'accéder à la valeur associée à la clé « house » ?

. EXERCICE 2 : CRÉATION PAR COMPRÉHENSION

1. Que fait cette ligne ? `d = { chr(65+i) : i for i in range(26) }`

.....

2. Créer un dictionnaire ayant pour clés les nombres entiers de 1 à 20 et pour valeurs les carrés des clés. Afficher ce dictionnaire pour vérifier son contenu.
3. Quelle instruction permet d'accéder à la longueur de ce dictionnaire ?
4. Quelle instruction permet d'accéder à la valeur associée à la clé 3 ?

II. EXERCICES D'APPLICATION

. EXERCICE 3 : CODAGE / DÉCODAGE

1. Créer par compréhension une liste `lst` contenant les lettres de l'alphabet.
Aide : adapter la ligne proposée à la question 1 de l'exercice 2 pour avoir une liste au lieu d'un dictionnaire.
2. Mélanger la liste `lst`
Aide : utiliser la fonction `shuffle(x)` du module `random`
3. Que fait cette ligne ? `code = { chr(65+i) : lst[i] for i in range(26) }`
.....

4. Que fait cette ligne ? `decode = { code[cle] : cle for cle in code.keys() }`
.....

- Vérifier que `decode[code["A"]]` renvoie bien la valeur "A".
- Créer une fonction `translate (mot , code)` qui renvoie le mot obtenu en remplaçant chaque lettre du mot par la valeur qui lui est associée dans le dictionnaire `code`.
- Créer une variable `test` contenant un mot (en majuscules) choisi par l'utilisateur.
Aide : utiliser la fonction input
- Créer une variable `c` contenant le mot renvoyé par `translate(test, code)`
- Afficher le contenu de la variable `c`.
- Vérifier que si vous faites un appel à la fonction `translate` en lui passant comme argument la variable `c` et le dictionnaire `decode`, elle renvoie bien le mot choisi au départ.

. EXERCICE 4 : BATAILLE AVEC UN JEU DE 52 CARTES

- Créer un dictionnaire `valeur_carte` dont les clés sont les noms des cartes {As, Roi, Dame, Valet, 10, 9, ..., 3, 2}. On considère que la valeur d'un as vaut 14, celle d'un roi vaut 13, et ainsi de suite en diminuant jusqu'à 2.
- Afficher `valeur_carte['Valet']` . Vérifier que l'on obtient 11.
- Que fait cette ligne ?

```
jeu_carte = [ (nom, 'coeur') for nom in valeur_carte.keys() ]
```


.....
- En utilisant l'opérateur de concaténation, compléter `jeu_carte` pour que le jeu soit complet.
- Si le module `random` est importé, que fait cette ligne ?

```
echantillon = sample( jeu_carte, 2)
```


.....
- Afficher les deux cartes de l'échantillon ainsi que leur valeurs.
- Comparer ces deux valeurs pour annoncer la carte gagnante.

III. EXERCICE EXTRAIT DU LIVRE « PREPA BAC : NSI PREMIÈRE », HATIER

. EXERCICE 5 PAGE 94 (FACULTATIF)

On modélise des informations (nom, taille et poids) sur des Pokémon de la façon suivante :

```
pokemons = { 'Bulbizarre' : ( 70 , 7 ), 'Herbizarre' : ( 100 , 13 ), 'Abo' : ( 200 , 7 ), 'Jungko' : ( 170 , 52 ) }
```

Par exemple, Bulbizarre est un Pokémon qui mesure 70 cm et qui pèse 7 kg.

- Quel est le type de la variable `pokemons` ?
- Quelle instruction permet d'ajouter à cette structure le pokemon Goupix qui mesure 60 cm et qui pèse 10 kg ?
- Recopier cette fonction et écrire son *docstring*.

```
def le_plus_grand(pokemons):
    grand, taille_max = None, None
    for ( nom, (taille, poids) ) in pokemons.items():
        if taille_max is None or taille > taille_max:
            grand, taille_max = nom, taille
    return (grand, taille_max)
```

- Quelle est la valeur de `le_plus_grand(pokemons)` ?
- Écrire le code d'une fonction `le_plus_léger` qui prend en paramètre un dictionnaire de Pokémon en paramètre et qui renvoie un tuple dont la première composante est le nom du Pokémon le plus léger et la deuxième composante est son poids.
- Vérifier que `le_plus_léger(pokemons)` renvoie ('Bulbizarre' , 7).
- Écrire le code d'une fonction `taille` qui prend en paramètre un dictionnaire de Pokémon ainsi que le nom d'un Pokémon, et qui renvoie la taille de ce Pokémon.
- En utilisant la commande `assert`, proposer un jeu de test pour cette fonction