

Interrogation : Dictionnaires

L'évaluation porte sur 3 exercices indépendants.

Les exercices sont notés sur 18 et la rigueur, rédaction et justifications sont notés sur 2 points.

Exercice 1 : Questions de cours (2 points)

1. Comment rajouter une clef dans un dictionnaire ? Donner l'instruction Python.
2. Créer le dictionnaire Python associé à cet objet :

pays	capitale	population	continent	langues officielles
Belgique	Bruxelles	11.5M	Europe	français, néerlandais, allemand

3. Donner l'instruction Python qui permet de modifier la population de 11.5M à 11.6M.

Exercice 2 : Complétion de programme (4 points)

Compléter la fonction `filtrer_dictionnaire_inf` qui prend en paramètre un dictionnaire et une valeur plafond, et renvoie un nouveau dictionnaire ne contenant que les éléments inférieurs à cette valeur.

Exemple :

```
filtrer_dictionnaire_inf({'chat': 8, 'chien': 12, 'hamster': 2, 'lapin': 5}, 6) # Renvoie: {'hamster': 2, 'lapin': 5}
filtrer_dictionnaire_inf({'a': 15, 'b': 20, 'c': 25}, 10) # Renvoie: {}
```

```
def filtrer_dictionnaire_inf(dico: dict, plafond: int) -> dict:
    resultat = {}
    for ... :
        if ... < plafond:
            resultat[...] = ...
    return resultat
```

Exercice 3 : Écriture de programmes (12 points)

Écrire une fonction `compter_lettres` qui prend une chaîne de caractères et renvoie un dictionnaire où les clés sont les lettres de la chaîne et les valeurs sont leurs nombres d'occurrences (sans tenir compte de la casse).

Exemple :

`compter_lettres("Bonjour le Monde")` doit renvoyer `{'b': 1, 'o': 2, 'n': 2, 'j': 1, 'u': 1, 'r': 1, 'l': 1, 'e': 2, 'm': 1, 'd': 1}`

Écrire une fonction `prix_par_categorie` qui prend un dictionnaire de prix et renvoie un nouveau dictionnaire classant les produits par gamme de prix.

Critères de catégories :

Moins de 20€ : "Premier prix"

Entre 20€ et 50€ : "Milieu de gamme"

Entre 50€ et 100€ : "Haut de gamme"

100€ et plus : "Luxe"

```
>>> prix = {'Stylo': 15, 'Sac': 45, 'Montre': 85, 'Tablette': 250}
>>> prix_par_categorie(prix)
{
    'Premier prix': ['Stylo'],
    'Milieu de gamme': ['Sac'],
    'Haut de gamme': ['Montre'],
    'Luxe': ['Tablette']
}
```

Écrire une fonction `score_min` qui prend un dictionnaire et renvoie le nom du joueur ayant le plus petit score.

Exemple :

`score_min({'Alice': 850, 'Bob': 920, 'Charlie': 740})` doit renvoyer `'Charlie'`

Écrire une fonction `fusionner_dicos` qui prend deux dictionnaires et renvoie un nouveau dictionnaire contenant toutes les paires clé-valeur des deux dictionnaires. Si une clé est présente dans les deux dictionnaires, la valeur du second dictionnaire est conservée.

Exemple :

`fusionner_dicos({'a': 1, 'b': 2}, {'b': 3, 'c': 4})` doit renvoyer `{'a': 1, 'b': 3, 'c': 4}`