

TP 2: Classe Compteur

Note : Ce TP nécessite une version de python supérieure ou égale à 3.9

L'objectif de ce TP est de développer une classe `Compteur` du module `compteur.py` qui permet de compter des éléments immuables, puis d'obtenir un certain nombre de statistiques sur ces éléments.

Cette classe proposera les méthodes suivantes :

- `__init__` qui permet d'initialiser le compteur, optionnellement à partir d'un dictionnaire d'éléments et du nombre d'occurrences de ces éléments ;
- `incrémenter` qui permet d'incrémenter le nombre d'occurrences d'un élément. Si celui-ci n'était pas référencé, son nombre d'occurrences est alors de 1 ;
- `fixer` qui permet de fixer le nombre d'occurrences d'un élément ;
- `nb_occurrences` qui permet d'obtenir le nombre d'occurrences d'un élément. Si ce dernier n'a jamais été renseigné, alors la valeur 0 est retournée ;
- `elements` qui permet d'obtenir l'ensemble de tous les éléments référencés ;
- `elements_moins_frequents` qui permet d'obtenir l'ensemble des éléments référencés les moins fréquents ;
- `elements_plus_frequents` qui permet d'obtenir l'ensemble des éléments référencés les plus fréquents ;
- `elements_par_nb_occurrences` qui permet d'obtenir un dictionnaire dont les clés sont les nombres d'occurrences des éléments référencés et les valeurs un ensemble de tous les éléments ayant ce nombre d'occurrences.

Enfin deux instances de `Compteur` sont égales s'ils ont des statistiques sur des éléments égales.

Vous veillerez à :

- avoir un code *pythonic* ;
- avoir un score de 10 avec pylint.

La fonction `main` du module, automatiquement exécutée lorsque le module est utilisé en tant que script, permet de tester les codes développés :

```
$ python compteur.py
nb_occurrences : OK
elements : OK
elements_moins_frequents : OK
elements_plus_frequents : OK
elements_par_nb_occurrences : OK
str : OK
repr : OK
égalité: OK
```