

Corrigé sujet **32** - Année : 2023

Sujet 32 - 2023 [↓](#)

Exercice 1

```
1 def min_et_max(tab):
2     mini, maxi = tab[0], tab[0]
3     for elt in tab:
4         if elt < mini: mini = elt
5         if elt > maxi: maxi = elt
6     return {'min' : mini, 'max' : maxi}
```



Commentaires

Encore une recherche classique de minimum et de maximum, le résultat est renvoyé sous la forme d'un dictionnaire.

Exercice 2

```
1 class Carte:
2     def __init__(self, c, v):
3         """ Initialise les attributs couleur (entre 1 et 4), et valeur
4         (entre 1 et 13). """
5         self.couleur = c
6         self.valeur = v
7
8     def get_valeur(self):
9         """ Renvoie la valeur de la carte : As, 2, ..., 10, Valet, Dame,
10        Roi """
11        valeurs = ['As', '2', '3', '4', '5', '6', '7', '8', '9', '10',
12        'Valet', 'Dame', 'Roi']
13        return valeurs[self.valeur - 1]
14
15    def get_couleur(self):
16        """ Renvoie la couleur de la carte (parmi pique, coeur, carreau,
17        trèfle). """
18        couleurs = ['pique', 'coeur', 'carreau', 'trèfle']
19        return couleurs[self.couleur - 1]
20
```

```

21 class Paquet_de_cartes:
22     def __init__(self):
23         """ Initialise l'attribut contenu avec une liste des 52 objets
24         Carte possibles
25             rangés par valeurs croissantes en commençant par pique, puis
26         coeur,
27             carreau et trèfle. """
28         self.contenu = [Carte(couleur,valeur) for couleur in range(1,5)
29                         for valeur in range(1,14) ] # ❶
30
31     def get_carte(self, pos):
32         """ Renvoie la carte qui se trouve à la position pos (entier
33         compris entre 0 et 51). """
34         assert 0 <= pos <= 51, "Paramètre pos non valide" # ❷
35         return self.contenu[pos] # ❸

```

1. On utilise une définition de liste par compréhension pour parcourir les couleurs et valeurs possibles. Attention les numéros de couleurs vont de 1 à 4 (inclus) (donc `range(1, 5)`) et les valeurs de 1 à 13 inclus (donc `range(1, 14)`)
2. C'est l'instruction `assert` demandée : on vérifie que le paramètre `pos` est correct.
3. Un jeu de cartes contient 52 cartes, la position d'une carte est entre 0 et 51 inclus.



Remarque

- Il serait pertinent d'utiliser un dictionnaire afin d'associer valeur et nom de d'une carte : {1 : 'As', 2:'2', ..., 12: 'Dame', 13 : 'Roi'}