

Devoir 2 - Programmation orientée objets

Un fabricant de brioches décide d'informatiser sa gestion des stocks. Il écrit pour cela un programme en langage Python. Une partie de son travail consiste à développer une classe `Stock` dont la première version est la suivante :

```
class Stock:
    def __init__(self):
        self.qt_farine = 0
        self.oeufs = 0
        self.qt_beurre = 0
```

QUESTION 1

Ecrire une méthode `ajouter_beurre(self, qt)` qui ajoute la quantité `qt` de beurre à un objet de la classe `Stock`.

On admet que l'on a écrit deux autres méthodes `ajouter_farine` et `ajouter_oeufs` qui ont des fonctionnements analogues.

QUESTION 2

Ecrire une méthode `afficher(self)` qui affiche la quantité de farine, d'oeufs et de beurre d'un objet de type `Stock`. L'exemple ci-dessous illustre l'exécution de cette méthode dans la console :

```
>>> mon_stock = Stock()
>>> mon_stock.afficher()
farine : 0
oeufs : 0
beurre : 0

>>> mon_stock.ajouter_beurre(560)
>>> mon_stock.afficher()
farine : 0
oeufs : 0
beurre : 560
```

Pour faire une brioche, il faut 350g de farine, 175g de beurre et 4 oeufs.

QUESTION 3

Ecrire une méthode `stock_suffisant_brioche(self)` qui renvoie un booléen VRAI s'il y a assez d'ingrédients dans la stock pour faire une brioche FAUX sinon.

On considère la méthode `produire(self)` de la classe `Stock` donnée par la code suivant :

```
def produire(self):
    res = 0
    while self.stock_suffisant_brioche():
        self.qt_beurre = self.qt_beurre - 175
        self.qt_farine = self.qt_farine - 350
        self.qt_oeufs = self.qt_oeufs - 4
        res = res + 1
    return res
```

On considère un stock défini par les instructions suivantes :

```
>>> mon_stock=Stock()  
>>> mon_stock.ajouter_beurre(1000)  
>>> mon_stock.ajouter_farine(1000)  
>>> mon_stock.ajouter_oeufs(10)
```

QUESTION 4

(a) On exécute l'instruction suivante :

```
>>>mon_stock.produire()
```

Que renvoie l'instruction ? Que représente cette valeur ?

(b) On exécute l'instruction suivante :

```
>>> mon_stock.afficher()
```

Que s'affiche-t-il dans la console ?

L'industriel possède n lieux de production distincts et donc n stocks distincts.

On suppose que ces stocks sont dans une liste dont chaque élément est un objet de type **Stock**.

QUESTION 5

Écrire une fonction Python `nb_brioche(liste_stocks)` possédant pour unique paramètre la liste des stocks et renvoie le nombre total de brioches produites.