

2026 sujet 6

Il faut, dans un premier temps, désactiver les tests en bas du fichier et les réactiver un par un à chaque question.

Question 1

```
def smoothie_possible(self, nom_smoothie):  
    """Retourne True si le smoothie peut être préparé  
    avec les fruits disponibles, False sinon."""  
    for fruit in self.db_smoothies[nom_smoothie]:  
        if fruit not in self.liste_fruits_disponibles:  
            return False  
    return True
```

Question 2

[Il ya une erreur dans le code. Dans la fonction `test_liste_smoothies_possibles` il faut remplacer `Berry Mix` par `Rouge kiwi`]

```
def liste_smoothies_possibles(self):  
    """Retourne la liste des smoothies pouvant être préparés  
    avec les fruits disponibles."""  
    liste = []  
    for nom_smoothie in self.db_smoothies:  
        if self.smoothie_possible(nom_smoothie):  
            liste.append(nom_smoothie)  
    return liste
```

Question 3

```
def test_score_proximité():  
    boutique = Boutique_smoothie([])  
    assert boutique.score_proximité("Tropical", "Rouge") == 0  
    assert boutique.score_proximité("Tropical", "Soleil couchant") == 1  
    assert boutique.score_proximité("Tropical", "Tropical citron") == 2  
    assert boutique.score_proximité("Tropical", "Tropical") == 3
```

Question 4

[Erreur dans le sujet, la fonction `plus_proche_possible` ne renvoie pas un dictionnaire.]

Le problème de la fonction `plus_proche_possible` est qu'elle renvoie le smoothie lui-même. Il faut donc s'assurer qu'il ne puisse pas être une possibilité :

```
def plus_proche_possible(self, nom_smoothie_ref):  
    """Retourne le nom du smoothie le plus proche de nom_smoothie_ref en  
    termes de fruits communs parmi les smoothies possibles."""
```

```

En cas d'égalité, retourne le premier trouvé.
"""
max_communs = 0
smoothie_proche = None
for nom_smoothie in self.db_smoothies:
    if nom_smoothie != nom_smoothie_ref :
        nb_communs = self.score_proximité(nom_smoothie_ref, nom_smoothie)
        if nb_communs > max_communs:
            max_communs = nb_communs
            smoothie_proche = nom_smoothie
return smoothie_proche

```

[Erreur dans le code, le deuxième test n'est pas correct, il faut le changer pour que les tests fonctionnent :]

```

def test_plus_proche_possible():
    boutique = Boutique_smoothie(
        ["Mangue", "Ananas", "Banane", "Fraise", "Citron", "Kiwi", "Pomme verte"])
    smoothie_proche = boutique.plus_proche_possible("Tropical")
    assert smoothie_proche == "Tropical citron"

    smoothie_proche2 = boutique.plus_proche_possible("Exotique")
    assert smoothie_proche2 == "Exotique rouge"

```

Question 5

Il suffit de décommenter les dernières lignes. Le smoothie proposé est Tropical citron