

## 2026 sujet 18

## Question 1

```
def temperature_moyenne(zone, donnees):
    nb = 0
    temp = 0
    for mesure in donnees:
        if mesure['zone'] == zone:
            nb += 1
            temp += mesure['temperature']
    if nb == 0:
        return None
    else:
        return temp / nb
```

## Question 2

```
def detecter_anomalies(zone, seuil, donnees):
    liste = []
    moy = temperature_moyenne(zone, donnees)
    for mesure in donnees:
        if mesure['zone'] == zone and abs(moy - mesure['temperature']) > seuil:
            liste.append(mesure['date'])
    return liste
```

## Question 3

```
def test_zone_inexistante():
    """
    Test 1 : Tester une zone qui n'existe pas

    À compléter:
    1. Appeler evolution_par_decennie avec une zone inexistante
    2. Vérifier que le résultat est un dictionnaire vide
    """
    assert evolution_par_decennie('Australes', donnees_test) == {}

def test_une_seule_decennie():
    """
    Test 2: Tester une zone avec données sur une seule décennie

    À compléter:
    1. Appeler evolution_par_decennie avec la zone appropriée
```

2. Vérifier que le résultat ne contient qu'une seule décennie (2020)

3. Vérifier la température moyenne

```
"""
```

```
assert len(evolution_par_decennie('Marquises', donnees_test)) == 1
```

```
print(evolution_par_decennie('Marquises', donnees_test))
```

```
assert evolution_par_decennie('Marquises', donnees_test)[2020] == 26.0
```

```
def test_plusieurs_decennies():
```

```
"""
```

*Test 3 : Tester une zone avec données sur plusieurs décennies*

*À compléter:*

1. Appeler *evolution\_par\_decennie* avec la zone appropriée

2. Vérifier que le résultat contient bien les clés 2010 et 2020

3. Vérifier que les températures moyennes sont cohérentes

```
"""
```

```
assert 2010 in evolution_par_decennie('Tuamotu', donnees_test)
```

```
assert 2020 in evolution_par_decennie('Tuamotu', donnees_test)
```

```
assert evolution_par_decennie('Tuamotu', donnees_test)[2010] == 26.8
```

```
assert evolution_par_decennie('Tuamotu', donnees_test)[2020] == 27.75
```

Les tests montrent que les clés ne sont pas bonnes.

#### Question 4

Il faut changer le calcul de la décennie et tout fonctionne :

```
# Calcul de la décennie
```

```
decennie = (annee // 10) * 10
```