

## 1.1 – Un exemple : le paradoxe des anniversaires

### Groupe 1 – les ensembles

Connais-tu le **paradoxe des anniversaires** ? Lorsqu'on réunit 23 personnes, il y a une chance sur deux pour que deux d'entre elles soient nées le même jour.

Ecrivons un programme qui vérifie expérimentalement ce paradoxe.

On va répéter de nombreuses fois un tirage aléatoire de 23 dates d'anniversaire, c'est-à-dire 23 entiers entre 1 et 365, on va observer combien de fois au moins deux personnes sont nées el même jour.

```
[2]: # Programme 1

def contient_doublon(t):
    """Le tableau t contient-il un doublon ?"""
    s = set()
    for x in t:
        if x in s:
            return True
        s.add(x)
    return False

# Création du tableau de dates aléatoires
from random import randint

n = 0
n_doublons = 0
while n > 1000:
    t = [None] * 23
    for i in range(23):
        t[i] = randint(1,365)
    if contient_doublon(t):
        n_doublons += 1
    n += 1

print(n_doublons, "doublons sur", n, "tirages")
print("fréquence : ", n_doublons/n)
```

501 doublons sur 1000 tirages  
fréquence : 0.501

*L'objectif de cette activité est de présenter la fonction `contient_doublon(t)` à vos camarades.* Afin de vous aider à vous approprier la problématique, voici une liste de questions :

### Analyse du programme 1 :

1. À quoi sert la boucle `while n < 1000` ?
2. Que pensez-vous de l'affirmation du paradoxe des anniversaires ?

### Utilité de la fonction `contient_doublon(t)` :

3. Dans la fonction `contient_doublon(t)`, quel est le type Python de la variable `t` ?
4. Que renvoie la fonction `contient_doublon(t)` ?
5. Comment fonctionne la ligne `if contient_doublon(t)` ?
6. Quel est l'intérêt de la fonction `contient_doublon(t)` ?

### Fonctionnement de la fonction `contient_doublon(t)` :

7. Quel est le type Python de la variable `s` ?
8. Quel est le but de la commande `s.add(x)` ?
9. Quel est l'effet de la ligne `for x in t` ?
10. L'instruction `if x in s` est-elle immédiate ?