

# Exercices sur les Bases de la programmation

# Nom complet

- Écrire une fonction qui prend en paramètre :
  - prenom
  - nom
- Elle retournera une chaîne avec le prénom et le nom séparés d'un espace, exemple : « John Doe »
- Vous afficherez ensuite le résultat de cette fonction

# Soustraction

- Écrire la fonction **soustraire** qui prend en paramètre:
  - nombre a
  - nombre b
- Elle retournera un entier qui sera la soustraction de ces deux nombres
  - exemple : `soustraire(2, 1)` => résultat = 1
- De plus, lors de l'exécution la fonction affichera « Je soustrait 2 et 1 »
- Vous afficherez ensuite le résultat de cette fonction

# Quelle Heure

- Écrire une fonction **quelle\_heure**
- Cette fonction aura un paramètre heure de type str
- Ce paramètre aura "12h00" comme valeur par défaut
- La fonction ne retournera aucun résultat mais affichera l'heure directement
- exemple : `quelle_heure()` => résultat : "il est 12h00"
- exemple : `quelle_heure("14h00")` => résultat : "il est 14h00"

# Compteur de lettre

- Écrire une fonction **compter\_lettre\_a**
- Cette fonction prendra en paramètre une chaîne
- Créer une boucle qui parcourt les lettres de la chaîne et compte le nombre de lettres égales à "a"
- La fonction renverra un entier
- exemple : `compter_lettre_a("C'est le b-a ba")` => résultat : 2
- exemple : `compter_lettre_a("mixer")` => résultat : 0
- Écrire une autre fonction sans boucle qui utilisera `count` à la place.

# ADN

Écrire un programme qui permet de saisir une chaîne d'ADN ainsi qu'une séquence d'ADN et qui retourne le % d'occurrences de la séquence dans la chaîne

Cette séquence sera composée uniquement de la combinaison de lettre suivante : 'a', 't', 'c', 'g'

1. Écrire une fonction **vérification\_adn** qui permet de renvoyer la valeur True si la chaîne d'ADN est valide, False si elle est invalide
2. Écrire une fonction **saisie\_adn** qui récupère une saisie, vérifie sa validité et renvoie une chaîne d'ADN valide sous forme de chaîne
3. Écrire une fonction **proportion** qui reçoit deux paramètres : une chaîne d'ADN et une séquence d'ADN. Elle renverra le % d'occurrences de la séquence dans la chaîne
4. Créer des instructions pour pouvoir tester le programme

# Tuple d'operations

- Écrire un programme se servant d'une fonction nommée **operation** retournant, à partir de deux nombres lui étant envoyés en paramètres :
  - La somme
  - La difference
  - Le quotient
  - Le produit
- Vous testerez cette fonction dans le cadre d'un programme console demandant à l'utilisateur deux valeurs et lui permettant d'obtenir les 4 résultats en même temps