



Exercices Python – Listes et calculs

Exercice 1 - Étendue (6206)

Écrire une fonction `etendue(liste)` qui retourne l'étendue (différence entre le max et le min) d'une liste.

Exercice 2 - Moyenne (3112)

Écrire une fonction `moyenne(liste)` qui retourne la moyenne des éléments de `liste`.

Exercice 3 - Médiane (3810)

Écrire une fonction `mediane(liste)` qui retourne la médiane des éléments de `liste`.

Exercice 4 - Diviseurs d'un entier (2486)

Écrire une fonction `diviseurs(n)` qui renvoie la liste des diviseurs positifs de `n`, dans l'ordre croissant.

```
>>> diviseurs(15)
[1, 3, 5, 15]
```

Exercice 5 - Doubles (1137)

Écrire une fonction `doubler(liste)` qui renvoie la liste des doubles des nombres de `liste`.

```
>>> doubler([1,2,3])
[2, 4, 6]
```

Exercice 6 - Somme des éléments d'une liste (739)

Écrire une fonction `somme(liste)` qui retourne la somme des éléments de `liste`.

⚠ La fonction `sum` est interdite.

```
>>> somme([1,2,3])
6
```

Exercice 7 - Produit des éléments d'une liste (699)

Écrire une fonction `produit(liste)` qui retourne le produit des éléments de `liste`.

```
>>> produit([1,2,3,4])
24
```

Exercice 8 - Nombre de pairs (489)

Écrire une fonction `nombrePairs(liste)` qui retourne, pour une liste de nombres entiers, le nombre d'entiers pairs de la liste.

```
>>> nombrePairs([1,2,3,6,11])
2
```

Exercice 9 - Nombre de négatifs (473)

Écrire une fonction `nombreNegatifs(liste)` qui retourne le nombre de nombres négatifs de la liste.

```
>>> nombreNegatifs([1,-2,3,-4,5])
2
```

Exercice 10 - Transformation (998)

Écrire une fonction `zeroSiNegatif(liste)` qui retourne `liste` (une liste d'entiers relatifs) modifiée de sorte que les éléments négatifs sont remplacés par 0.

```
>>> zeroSiNegatif([1,-2,6,-3,-89])
[1, 0, 6, 0, 0]
```

Exercice 11 - Moyenne quadratique (139)

La moyenne quadratique de nombres (x_1, x_2, \dots, x_n) est définie par :

$$[MQ = \sqrt{\frac{x_1^2 + x_2^2 + \dots + x_n^2}{n}}]$$

Écrire la fonction `moyenneQuadratique(liste)` qui retourne la moyenne quadratique des éléments de `liste` arrondie à 4 décimales.

```
>>> moyenneQuadratique([1,2,3])
2.1602
```

Exercice 12 - Moyenne harmonique (401)

La moyenne harmonique de nombres strictement positifs (x_1, x_2, \dots, x_n) est définie par :

$$[MH = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + \dots + \frac{1}{x_n}}]$$

Écrire la fonction `moyenneHarmonique(liste)` qui retourne la moyenne harmonique des éléments de `liste` arrondie à 4 décimales.

Exercice 13 - Suite de Syracuse (186)

La suite de Syracuse est définie ainsi :

- Si le nombre est pair, on le divise par 2
- Si le nombre est impair, on le multiplie par 3 et on ajoute 1
- On recommence jusqu'à atteindre 1

Écrire une fonction `syracuse(N)` qui retourne la liste des nombres obtenus dans la suite de Syracuse jusqu'à rencontrer 1.

```
>>> syracuse(5)
[5, 16, 8, 4, 2, 1]
```

Exercice 14 - Suite de Fibonacci (1187)

La suite de Fibonacci est définie par : $[F_0 = 0, \quad F_1 = 1, \quad F_n = F_{n-1} + F_{n-2}]$

Écrire une fonction `fibonacci(n)` qui renvoie la liste des `n` premiers nombres de Fibonacci.

Exercice 15 - L'important c'est d'essayer (916)

Écrire des fonctions `premier(participants, classement)` et `dernier(participants, classement)`, retournant respectivement le premier arrivé et le dernier arrivé dans une course de vélo.

- `participants` est une liste de prénoms (participants à la course)
- `classement` est une liste d'entiers (positions dans la course)
- Les deux listes ont la même longueur

```
>>> participants = ["Alice", "Bob", "Charlie", "David"]
>>> classement = [2,4,1,3]
>>> premier(participants, classement)
'Charlie'
>>> dernier(participants, classement)
'Bob'
```

Exercice 16 - Inverser une liste (855)

Écrire une fonction `inverser(liste)` qui retourne `liste` inversée.

```
>>> inverser([1,2,3,4])  
[4, 3, 2, 1]
```