

PYTHON № 2 : DÉFINIR UNE FONCTION

1 ex simple : Indice de Masse Corporelle

```
1 # Définition de la fonction indice(T,P)
2 def indice(T,P):
3     return int(round(P/T**2,0))
4
5 #Utilisation cette fonction
6 print("Quelle est votre taille , en m ?")
7 taille=eval(input())
8 print("Quelle est votre poids , en kg ?")
9 poids=eval(input())
10 print("Voici votre indice de masse corporelle : ",indice(taille,poids))
```

2 amélioration de l'ex

```
1 # Définition de la fonction indice(T,P)
2 def indice(temps, poids = 70):
3     return int(round(poids/T**2,0))
4
5 # Utilisation :
6 # 1/ la valeur par défaut du poids est fixé à 70 kg (si pas de valeur)
7 # 2/ les paramètres nommés peuvent être rentrer dans l'ordre désiré
```

3 ex plus sophistiqué : conversion du temps

```
1 # Conversion des secondes en heures, minutes et secondes
2 def conversion(t):
3     t=int(t)
4     H=t//3600
5     m=t%3600
6     M=m//60
7     S=m%60
8     return H,M,S                                # type tuple
9
10 # Utilisation de la fonction "conversion"
11 print("Quelle durée, en secondes, voulez-vous convertir ?")
12 temps=eval(input())
13 print(temps," secondes font ",conversion(temps))
```

4 Return n'est pas obligatoire : parité d'un nombre

```
1 # définition de la fonction "pair"
2 def pair(n):
3     if n%2==0:
4         print(n, " est pair")
5     else:
6         print(n, " est impair")
7
8 # Utilisation de la fonction "pair"
9 n=eval(input("Choisissez un entier : "))
10 pair(n)
```

5 utiliser une fonction stockée dans un fichier

fichier contenant les fonctions : mesfonctions.py

```
1 # définition de la fonction "pair"
2 def pair(n):
3     if n%2==0:
4         print(n, " est pair")
5     else:
6         print(n, " est impair")
```

fichier principal : main.py

```
1 # Importer la fonction stockée
2 import mesfonctions
3
4 # Utilisation de la fonction "pair"
5 n=eval(input("Choisissez un entier : "))
6 mesfonctions.pair(n)
```

Remarque : pour importer les fonctions pré-existantes, écrire : `from math import*`

6 fonction à usage unique : lambda

lambda en 2 temps

```
1 # définition de la fonction f(x) = 2*x+y-1
2 f = lambda x,y: 2*x+y-1
3 print(f(5,3))
```

lambda directement

```
1 # utilisation directe de la fonction f(x) = 2*x+y-1
2 print((lambda x,y: 2*x+y-1)(5,3))
```

7 Exercice

Ex 1 : créer 1 fonction qui demande 1 rayon et qui calcule périmètre et aire du cercle

Ex 2 : créer une fonction fibo qui calcule le terme de rang n de la suite de Fibonacci

Ex 3 : choisir un entier n de 3 chiffres ; définir 1 fonction qui retourne m en prenant les chiffres de n dans le sens inverse ; existe-t-il 1 entier pour lequel on a : $n = 3m + 1$

Ex 4 : 1 nombre est dit parfait s'il est égal à la somme de ses diviseurs (excepté lui-même) : 6 est parfait car $6 = 1 + 2 + 3$; chercher les nombres parfaits de 1 à 1000