

Entrée [2]:

```
#Ecrire une fonction qui calcule l'indice de masse corporelle (IMC) d'un adulte et qui en d
#l'interprétation (corpulence normale, surpoids...).
taille=float(input("entrez votre taille en cm "))
poids=float(input("Entrez votre masse en kilogramme "))
def calc_imc(t,m):
    #conversion de la taille
    tail=t/100
    # conversion de la masse
    mass=m
    imc=round((mass/(tail*tail)),1)
    if(imc<18.5):
        message="Vous êtes maigre"
    elif (imc>=18.5 and imc<25):
        message="Votre corpulence est normale"
    elif (imc>=25 and imc<30):
        message="Vous êtes en surpoids"
    elif (imc>=30 and imc<35):
        message="Votre obésité est modérée"
    elif (imc>=35 and imc<40):
        message="Vous êtes obèse"
    elif (imc>=40):
        message="Votre obésité est massive ou morbide, faite attention à votre alimentation"
    print("votre IMC est ",imc,"kg/m² ",message)
calc_imc(taille,poids)
```

entrez votre taille en cm 165  
 Entrez votre masse en kilogramme 61  
 votre IMC est 22.4 kg/m² Votre corpulence est normale

Entrée [3]:

```
#Ecrire une fonction qui reçoit la note au bac et qui affiche la mention correspondante
def mention(note):
    mantion=""
    if(note<10):
        mantion=" recalé"
    elif (note>=10 and note<12):
        mantion="passable"
    elif (note>=12 and note<14):
        mantion="Assez bien"
    elif (note>=14 and note<16):
        mantion="Bien"
    elif (note>=16 and note<18):
        mantion="Très bien"

    print("avec la note ",note," vous avez obtenu la mention: ", mantion)
mention(17)
```

avec la note 17 vous avez obtenu la mention: Très bien

Entrée [6]:

```

#Ecrire une fonction qui retourne l'aire de la surface d'un disque de rayon R.
Prendre un paramètre qui précise l'unité de mesure :
import pi
print("Entrer cm pour centimetre et m pour metre: ")
rayon=float(input("entrer la valeur du rayon: "))
def aire_disque(rayon):
    aire=float(rayon*rayon)*pi
    print("l'aire de la surface d'un disque de rayon ",rayon,unite," est ",round(aire,2),unite, "²")
aire(Le_rayon)

```

```

Entrer cm pour centimetre et m pour metre: m
entrer la valeur du rayon: 5
l'aire de la surface d'un disque de rayon  5.0 m  est  78.54 m ²

```

Entrée [7]:

```

#Ecrire une fonction qui retourne la factorielle d'un nombre entier N
nb=int(input("entrez un nombre: "))
def factorielle(N):
    factoriel=1;
    for x in range(1, N+1):
        factoriel=factoriel*x
    print(N,"!=" ,factoriel)
factorielle(nb)

# ou avec une importation
from math import *
print(nb,"!=" ,factorial(nb))

```

```

entrez un nombre: 5
5 != 120
5 != 120

```

Entrée [8]:

```

#Ecrire une fonction qui reçoit un paramètre N et qui nous retourne la liste des nombres
#impairs compris entre 0 et N.
#Exemple : pour N = 10 la fonction nous retourne [1, 3, 5, 7, 9]
#Indication : utiliser la boucle while
nb=int(input("entrer un nombre: "))
def nombre_impair(N):
    a=0
    liste=[]
    while a<N:
        if((a%2)!=0):
            liste.append(a)
        a=a+1
    print("la liste des nombres impairs entre 0 et ",N," est ",liste)
nombre_impair(nb)

```

```

entrer un nombre: 13
la liste des nombres impairs entre 0 et  13  est  [1, 3, 5, 7, 9, 11]

```

Entrée [9]:

```
#Définir la liste : liste =[17, 38, 10, 25, 72], puis effectuez les actions suivantes :
liste=[17,38,10,25,72]
print("liste de départ ",liste)
#trier la liste avec sort()
liste.sort()
print("liste apres un triage ",liste)
# ajout de l'element 12
liste.append(12)
print("liste après ajout de l'élément 12 ",liste)
# renverser les element de la liste
liste.reverse()
print("liste après inversement des éléments ",liste)
# affichage du nombre d'élément de la liste
nb_element=len(liste)
print("le nombre d'élément de la liste est: ",nb_element)
#suppression de l'élément 38 de la liste
liste.remove(38)
print("liste après suppression de l'élément 38 ",liste)
# affichage de la sous liste du 2e et 3e element
print("la sous liste du 2e au 3e est ",liste[1:3])
# affichage de la sous liste du debut au 2e element
print("la sous liste du debut au 2e est ",liste[:2])
# affichage de la sous liste du 3e element jusqu'a la fin
print("la sous liste du 3e element jusqu'a la fin est ",liste[2:])
```

```
liste de départ  [17, 38, 10, 25, 72]
liste apres un triage  [10, 17, 25, 38, 72]
liste après ajout de l'élément 12  [10, 17, 25, 38, 72, 12]
liste après inversement des éléments  [12, 72, 38, 25, 17, 10]
le nombre d'élément de la liste est:  6
liste après suppression de l'élément 38  [12, 72, 25, 17, 10]
la sous liste du 2e au 3e est  [72, 25]
la sous liste du debut au 2e est  [12, 72]
la sous liste du 3e element jusqu'a la fin est  [25, 17, 10]
```

Entrée [10]:

```
#Ecrire une fonction qui supprime les doublons d'une liste.
liste1=[1, 1, 2, 4, 9, 2, 5, 4]
def sup_doublon(listes):
    liste=listes
    nb_doublon=0;
    liste_traite=[]
    for x in liste:
        if (x in liste_traite):
            nb_doublon=nb_doublon+1
        else:
            liste_traite.append(x)

    print("la liste initiale est: ",liste)
    print("Il y a ",nb_doublon, " doublons dans cette liste")
    print("après suppression des doublons ",liste_traite)
sup_doublon(liste1)
```

la liste initiale est: [1, 1, 2, 4, 9, 2, 5, 4]  
Il y a 3 doublons dans cette liste  
après suppression des doublons [1, 2, 4, 9, 5]

Entrée [ ]: