

0 Rappels 2 Listes fonctions

July 26, 2024

```
[1]: li2 = [n for n in range(300) if n%3==0]
```

```
[2]: len(li2)
```

```
[3]: li3 = [2*p+1 for p in range(30)]
```

```
[4]: print(li3)
```

```
[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59]
```

```
[5]: li = []  
for i in range(10):  
    li.append(i*i)
```

```
[6]: print(li)
```

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

```
[7]: for i in range(len(li)):  
    li[i] += 1
```

```
[8]: print(li)
```

```
[1, 2, 5, 10, 17, 26, 37, 50, 65, 82]
```

```
[9]: from math import sin, cos
```

```
[10]: li_sin2 = [sin(n)**2 for n in range(10)]  
li_cos2 = [cos(n)**2 for n in range(10)]
```

```
[11]: li_sum = [li_sin2[i] + li_cos2[i] for i in range(len(li_sin2))]
```

```
[12]: print(li_sum)
```

```
[1.0, 1.0, 1.0, 0.9999999999999999, 1.0, 1.0, 1.0, 0.9999999999999999, 1.0, 0.9999999999999999]
```

```
[13]: def carre(x:float)->float:  
    """
```

```
Retourne le carré du nombre x.
"""
return x**2
```

```
[14]: print(carre(3))
```

9

```
[15]: from math import sqrt
```

```
[16]: def distance(A:(float, float), B:(float, float))->float:
      """
      Retourne la distance euclidienne entre les points du plan A et B.
      """
      xA, yA = A
      xB, yB = B
      return sqrt((xB - xA)**2 + (yB - yA)**2)
```

```
[17]: print(distance((1, 0), (0, 1)))
```

1.4142135623730951

```
[18]: def valAbs(x:float)->float:
      """
      Retourne la valeur absolue du nombre x.
      """
      if x < 0:
          return -x
      else:
          return x
```

```
[19]: print(valAbs(-3))
```

3

```
[20]: def dman(A:(float, float), B:(float, float))->float:
      """
      Retourne la distance de Manhattant entre les points du plan A et B.
      """
      xA, yA = A
      xB, yB = B
      return valAbs(xB - xA) + valAbs(yB - yA)
```

```
[21]: dman((1, 0), (0, 1))
```

```
[22]: def affiche_liste(li:list)->None:
      """
      Affiche les éléments de la liste li et leur indices au format indice :_
      ↪élément
```

```
"""
for i in range(len(li)):
    print(i, ':', li[i])
```

```
[23]: affiche_liste([4, 5 ,6])
```

```
0 : 4
1 : 5
2 : 6
```

```
[24]: def zeros(n:int)->[int]:
      """
      Retourne une liste qui contient n fois zéro.
      """
      return [0] * n
```

```
[25]: print(zeros(3))
```

```
[0, 0, 0]
```

```
[26]: def zeros2D(n:int, m:int)->[[int]]:
      """
      Retourne une liste qui contient n listes qui contiennent m fois
      zéro chacune.
      """
      return [zeros(m) for i in range(n)]
```

```
[27]: print(zeros2D(2, 3))
```

```
[[0, 0, 0], [0, 0, 0]]
```

```
[28]: def reste(a:int, b:int)->int:
      """
      Retourne le reste de la division euclidienne de a par b.
      """
      while a >= b:
          a = a - b
      return a
```

```
[29]: print(reste(10, 3))
```

```
1
```

```
[30]: def quotient(a:int, b:int)->int:
      """
      Retourne le quotient de la division euclidienne de a par b.
      """
      c = 0
      while a >= b:
```

```
    a = a - b
    c += 1
    return c
```

```
[31]: print(quotient(10, 3))
```

3

```
[ ]:
```