

_9generalas/_5functions.py

```

1
2 """-----
   -----"""
3 doble = lambda x: x * 2
4 print("El doble de 5 es:", doble(5))
5
6 #sumar
7 def sumar (a,b):
8     """
9     Calcula la suma de dos argumentos.
10
11     Args:
12         a(parametro)
13         b(parametro)
14
15     Retorna:
16         int: La suma de a y b.
17
18     """
19     return(a+b)
20
21 print(sumar(5,7))
22
23 def suma(*args):
24     if all(isinstance(num, (int, float)) for num in args):
25         return sum(args)
26     else:
27         return "Not all list items are numbers"
28
29 #-----suma numeros-----
30 def sum_of_numbers(n):
31     return sum(range(n + 1))
32
33 print(sum_of_numbers(5))                # 15 (0 + 1 + 2 + 3 + 4 + 5 = 15)
34
35 print(sum([1, 2, 3, 4, 5]))              #15 Esto hace lo mismo que toda la funcion
36
37 list=[1, 2, 3, 4, 5]
38 def sumador(list):
39
40     print(sum([1, 2, 3, 4, 5]))           #15
41     print(sum(list))                     #15
42     suma=0
43     for numero in list:
44         suma+=numero
45
46     print(suma)                          #15
47
48 sumador(list)
49
50 #Global ?????
51 contador = 0

```

```
52 def incrementa_contador():
53     global contador
54     contador += 1
55 incrementa_contador()
56 print("El valor del contador es:", contador)
57
58 #externa????
59 def externa():
60     x = 10
61     def interna():
62         nonlocal x
63         x += 1
64         print("Valor de x dentro de la función interna:", x)
65     interna()
66     print("Valor de x fuera de la función interna:", x)
67 print("Llamando a la función externa...")
68 externa()
69
70 #Asegurar no es negativo???
71 x = 5
72 assert x > 0, "x no puede ser negativo"
73 print("La aserción pasó correctamente.")
74
75 #-----Even nums-----
76 def evenum(numm):
77     return numm%2==0
78 print(evenum(7))                                #False
79
80 #-----imprimir central.-----
81 lista=["champú", "colageno", "tomillo", "cardamomo", "Cura"]
82 numerin=len(lista)
83 print(numerin)
84 odd=int(numerin/2)
85 centro_even1= int(numerin/2-1)
86 centro_even2= int(numerin/2)
87
88 if len(lista)%2==0:
89     print(lista[centro_even1:centro_even2])
90 else:
91     print(lista[odd])
92
93 print(f" primera {lista[0]}, última {lista[-1]}, centro {lista[centro_even1]},
94 {lista[centro_even2]}")
95
96 """Try, except finally raise yield"""
97 #Gestión de Excepciones:
98 try:
99     resultado = 10 / 0
100 except ZeroDivisionError:
101     print("Error: División por cero")
102 finally:
103     print("Finalizando el bloque try-except")
104
```

```
105 try:
106     x = 1 / 0 #Inicia un bloque de código donde se
manejan las excepciones.
107 except ZeroDivisionError:
108     print("División por cero") #Captura y maneja una excepción
específica.
109     """raise ValueError("Valor inválido") #Genera una excepción específica."""
110
111
112 assert x > 0, "x debe ser mayor que cero"#Verifica si una condición es verdadera y genera
una excepción si no lo es.
113
114 primernumero=int(input("Introduzca el primer número: "))
115 segundonumero=int(input("Introduzca el segundo número: "))
116
117
118 def es_primo(numero):
119     for n in range(2,numero):
120         if numero % n ==0:
121             return False
122     print(str(numero)+ "es primo.")
123     return True
124
125 for i in range(primernumero,segundonumero):
126     es_primo(i)
127
128 def es_primo(numero):
129     if numero <= 1:
130         return False
131     for i in range(2, int(numero**0.5) + 1):
132         if numero % i == 0:
133             return False
134     return True
135
136 def imprimir_primos_entre(num1, num2):
137     print(f"Números primos entre {num1} y {num2}:")
138     for numero in range(num1, num2 + 1):
139         if es_primo(numero):
140             print(numero, end=" ")
141
142 # Pedir al usuario que introduzca dos números
143 numero1 = int(input("Introduce el primer número: "))
144 numero2 = int(input("Introduce el segundo número: "))
145
146 # Asegurarse de que el primer número sea menor o igual que el segundo número
147 if numero1 > numero2:
148     numero1, numero2 = numero2, numero1
149
150 # Imprimir los números primos entre los dos números introducidos
151 imprimir_primos_entre(numero1, numero2)
152
153 lista1=[]
154 lista2=[]
155 def comparalistas():
```

```
156
157     for i in lista1:
158         for j in lista2:
159             if i == j :
160                 cierto=True
161             else: cierto=False
162
163     if cierto==False:
164         print("diferentes")
165     else: print("igualitas uwu")
166
167 lista1=["Ana", "Me", "Ama"]
168 lista2=["Ana", "Me", "ama"]
169 comparalistas()
170
```