**Solución a los problemas del día 5: a las Prácticas del Día 5**

Abierto

Prácticas del Día 5

**Soluciones a las Prácticas del Día 5**

Soluciones a las Prácticas del Día 5

Antes de dar por finalizadas las prácticas del día de hoy y pasar al proyecto, te comparto las posibles soluciones para cada una de las prácticas que fueron propuestas el día de hoy, en caso de que hayas tenido alguna dificultad para terminarlas.

**¡Ten en cuenta que estas soluciones no son únicas!**Si has arribado por otro camino a un código que arroja un resultado correcto en PyCharm o tu editor de código de preferencia, quiero que sepas que has hecho un buen trabajo (no siempre pueden contemplarse todos los casos a la hora de evaluar una tarea).

**Práctica Métodos y Ayuda 1**

1. print(",:\_#,,,,,,:::\_\_\_\_##Pyt%on\_ \_Total,,,,,,::#".lstrip(",:%\_#"))

**Práctica Métodos y Ayuda 2**

1. frutas = ["mango", "banana", "cereza", "ciruela", "pomelo"]
2. frutas.insert(3,"naranja")

**Práctica Métodos y Ayuda 3**

1. marcas\_smartphones = {"Samsung", "Xiaomi", "Apple", "Huawei", "LG"}
3. marcas\_tv = {"Sony", "Philips", "Samsung", "LG"}
5. conjuntos\_aislados = marcas\_tv.isdisjoint(marcas\_smartphones)

**Práctica Crear Funciones 1**

1. def saludar():
2. print("¡Hola mundo!")

**Práctica Crear Funciones 2**

1. nombre\_persona = "Luis"
3. def bienvenida(nombre\_persona):
4. print(f'¡Bienvenido {nombre\_persona}!')

**Práctica Crear Funciones 3**

1. un\_numero = 5
3. def cuadrado(un\_numero):
4. print(un\_numero\*\*2)

**Práctica Return 1**

1. def potencia(num1, num2):
2. return num1\*\*num2

**Práctica Return 2**

1. dolares = 1200
3. def usd\_a\_eur(dolares):
4. return dolares\*0.90

**Práctica Return 3**

1. palabra = "Curso de Python"
3. def invertir\_palabra(palabra):
4. palabra = palabra[::-1]
5. palabra = palabra.upper()
6. return palabra

**Práctica Funciones Dinámicas 1**

1. lista\_numeros = [1,-50,502,-5000,755,600,33,61]
3. def todos\_positivos(lista\_numeros):
4. for numero in lista\_numeros:
5. if numero < 0:
6. return False
7. else:
8. pass
9. return True

**Práctica Funciones Dinámicas 2**

1. lista\_numeros = [1,50,500,5000,750,600]
3. def suma\_menores(lista\_numeros):
4. suma=0
5. for numero in lista\_numeros:
6. if numero in range(1,1000):
7. suma += numero
8. else:
9. pass
10. return suma

**Práctica Funciones Dinámicas 3**

1. lista\_numeros = [1,50,502,5000,755,600,33,61]
3. def cantidad\_pares(lista\_numeros):
4. cantidad=0
5. for numero in lista\_numeros:
6. if numero % 2 == 0:
7. cantidad += 1
8. else:
9. pass
10. return cantidad

**Práctica sobre Interacción entre Funciones 1**

1. import random
3. def lanzar\_dados():
4. return random.randint(1,6), random.randint(1,6)
6. def evaluar\_jugada(dado1, dado2):
7. suma\_dados = dado1 + dado2
8. if suma\_dados <= 6:
9. return f"La suma de tus dados es {suma\_dados}. Lamentable"
10. elif suma\_dados > 6 and suma\_dados < 10:
11. return f"La suma de tus dados es {suma\_dados}. Tienes buenas chances"
12. else:
13. return f"La suma de tus dados es {suma\_dados}. Parece una jugada ganadora"

**Práctica sobre Interacción entre Funciones 2**

1. lista\_numeros = [1,2,15,7,2,8]
3. def reducir\_lista(lista):
4. lista = list(set(lista))
5. lista.sort()
6. lista.pop(-1)
7. return lista
9. def promedio(lista):
10. valor\_medio = sum(lista)/len(lista)
11. return valor\_medio

**Práctica sobre Interacción entre Funciones 3**

1. lista\_numeros = [1,2,15,7,2,8]
3. import random
5. def lanzar\_moneda():
6. resultado = random.choice(["Cara", "Cruz"])
7. return resultado
9. def probar\_suerte(moneda, lista):
10. if moneda == "Cara":
11. print("La lista se autodestruirá")
12. return []
13. elif moneda == "Cruz":
14. print("La lista fue salvada")
15. return lista

**Práctica sobre Argumentos Indefinidos (\*args) 1**

1. def suma\_cuadrados(\*args):
2. suma = 0
3. for arg in args:
4. suma += arg\*\*2
6. return suma

**Práctica sobre Argumentos Indefinidos (\*args) 2**

1. def suma\_absolutos(\*args):
2. suma = 0
3. for arg in args:
4. suma += abs(arg)
6. return suma

**Práctica sobre Argumentos Indefinidos (\*args) 3**

1. def numeros\_persona(nombre, \*args):
2. suma\_numeros = sum(args)
3. return f'{nombre}, la suma de tus números es {suma\_numeros}'

**Práctica sobre Argumentos Indefinidos (\*kwargs) 1**

1. def cantidad\_atributos(\*\*kwargs):
2. cantidad = 0
3. for clave in kwargs.items():
4. cantidad += 1
5. return cantidad

**Práctica sobre Argumentos Indefinidos (\*kwargs) 2**

1. def lista\_atributos(\*\*kwargs):
2. lista = []
3. for valor in kwargs.values():
4. lista.append(valor)
5. return lista

**Práctica sobre Argumentos Indefinidos (\*kwargs) 3**

1. def describir\_persona(nombre, \*\*kwargs):
2. print(f"Características de {nombre}:")
3. for clave, valor in kwargs.items():
4. print(f'{clave}: {valor}')

Recuerda que si tienes alguna duda que aún no quede resuelta, tienes a tu disposición el espacio de Preguntas y Respuestas.

Te veo en el proyecto!