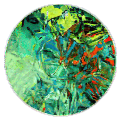




Fecha:10/12/24

## Índice:

1. Prueba escrita.....	2
2. Posible solución.....	4
3. Comentarios.....	6
4. Resultados.....	12



# 1. Prueba escrita

## 1. Variables y Tipos de Datos:

[1 punto] 1.1 Escribe un programa en Python que solicite al usuario su nombre y edad, y luego muestre un mensaje en pantalla con el formato: "Hola, [nombre]. Tienes [edad] años."

	<pre>¿Como te llamas? Ejemplo ¿Edad? 23 Hola, Ejemplo Tienes 23 años.</pre>
--	---

[1 punto] 1.2 Define las siguientes variables con un identificador correcto en Python con los valores indicados:

```
# Una variable de tipo entero con el valor 25.

# Una variable de tipo flotante con el valor 3.14.

# Una variable de tipo cadena con el valor "Python".
```

## 2. Control de Flujo – Condicionales

[1. punto] 2.1. Escribe un programa que solicite un número al usuario y verifique si es positivo, negativo o cero. Muestra el resultado correspondiente.

```
# Solicitar un número al usuario
numero = _____("Introduce un número: ")

# Verificar si el número es positivo, negativo o cero
if numero _____ 0:
    print("El número es _____.")
elif numero _____ 0:
    print("El número es _____.")
else:
    print("El número es _____.")
```

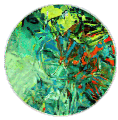
[2. puntos] 2.2. Escribe un programa que solicite dos números al usuario y determine cuál es el mayor, cuál es el menor o si son iguales.

## 3. Control de Flujo – Bucles

[1. punto] 3.1. Escribe un programa que imprima los números del 1 al 10 usando un bucle for.

```
# Usar un bucle for para imprimir números del 1 al 10
for i in _____(_____, ____):
    print(_____)
```

[2 puntos] 3.2 Escribe un programa que solicite un número entero positivo al usuario y calcule la suma de todos los números desde 1 hasta el número ingresado, usando un bucle while.



#### 4. Funciones

[1. punto] 4.1 .Implementa la función **acumulador**. Dicha función se le pasa una lista de enteros como parámetro de entrada y devuelve la suma total de todos los elementos que contiene.

```
# Definir la función acumulador
def acumulador(_____):
    # Inicializar la variable para almacenar la suma
    total = _____

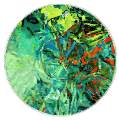
    # Recorrer la lista
    for _____ in _____:
        # Acumular el valor en total
        total += _____

    # Devolver el resultado
    _____ total
```

```
>>> acumulador([1,2,3])
6
>>> acumulador([])
0
>>> acumulador([3,10,6.5])
19.5
```

[1. punto] 4.2. Escribe una función llamada **suma** que reciba dos números como parámetros de entrada y devuelva como resultado la suma de ambas entradas. Asegúrate de que la función sea reutilizable y no incluye interacciones con el usuario directamente.

```
>>> suma(3,2)
5
>>> suma(10,4)
14
```



## 2. Posible solución

### 1. Variables y Tipos de Datos:

[1 punto] 1.1 Escribe un programa en Python que solicite al usuario su nombre y edad, y luego muestre un mensaje en pantalla con el formato: **"Hola, [nombre]. Tienes [edad] años."**

```
nombre = input("¿Cómo te llamas?")
edad = int(input("¿Edad?"))
print("Hola, " + nombre + " tienes " + str(edad) + " años.")
```

¿Como te llamas? **Ejemplo**  
¿Edad? **23**  
Hola, Ejemplo Tienes 23 años.

[1 punto] 1.2 Define las siguientes variables con un identificador correcto en Python con los valores indicados:

```
# Una variable de tipo entero con el valor 25.
edad = 25
# Una variable de tipo flotante con el valor 3.14.
pi = 3.14
# Una variable de tipo cadena con el valor "Python".
lenguaje = "Python"
```

### 2. Control de Flujo – Condicionales

[1. punto] 2.1. Escribe un programa que solicite un número al usuario y verifique si es positivo, negativo o cero. Muestra el resultado correspondiente.

```
# Solicitar un número al usuario
numero = int(input("Introduce un número: "))

# Verificar si el número es positivo, negativo o cero
if numero > 0:
    print("El número es positivo.")
elif numero == 0:
    print("El número es cero.")
else:
    print("El número es negativo.")
```

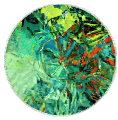
[2. puntos] 2.2. Escribe un programa que solicite dos números al usuario y determine cuál es el mayor, cuál es el menor o si son iguales.

```
# Solicitar un número al usuario
numero_1 = int(input("Introduce un número: "))
# Solicitar otro número al usuario
numero_2 = int(input("Introduce otro número: "))
# Verificar si el número es positivo, negativo o cero
if numero_1 > numero_2:
    print("El número ", numero_1, "es mayor que", numero_2)
elif numero_1 == numero_2:
    print("Los dos números son iguales.")
else:
    print("El número ", numero_2, "es mayor que", numero_1)
```

### 3. Control de Flujo – Bucles

[1. punto] 3.1. Escribe un programa que imprima los números del 1 al 10 usando un bucle for.

```
# Usar un bucle for para imprimir números del 1 al 10
for i in range(1, 11):
    print(i)
```



[2 puntos] 3.2 Escribe un programa que solicite un número entero positivo al usuario y calcule la suma de todos los números desde 1 hasta el número ingresado, usando un bucle `while`.

```
# Leemos el número positivo por teclado
print("Dime un número positivo")
numero = int(input())

# 'suma_total' es una variable que nos permitirá acumular la suma de todos los números
suma_total = 0
# 'i' es una variable que nos permite generar el número actual
i = 0

# Realizamos un bucle
# para recorrer todos los números y acumularlos en 'suma_total'
while(i < numero):
    i = i + 1
    suma_total = suma_total + i

print("La suma total desde 1 hasta", numero, " es ", suma_total)
```

**Nota:** Confiamos en que el usuario añade un número positivo por teclado.

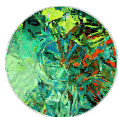
#### 4. Funciones

[1. punto] 4.1 Implementa la función **acumulador**. Dicha función se le pasa una lista de enteros como parámetro de entrada y devuelve la suma total de todos los elementos que contiene.

<pre># Definir la función acumulador def acumulador(lista):     # Inicializar la variable para almacenar la suma     total = 0      # Recorrer la lista     for numero in lista:         # Acumular el valor en total         total += numero      # Devolver el resultado     return total</pre>	<pre>&gt;&gt;&gt; acumulador([1,2,3]) 6 &gt;&gt;&gt; acumulador([]) 0 &gt;&gt;&gt; acumulador([3,10,6.5]) 19.5</pre>
---	--

[1. punto] 4.2. Escribe una función llamada **suma** que reciba dos números como parámetros de entrada y devuelva como resultado la suma de ambas entradas. Asegúrate de que la función sea reutilizable y no incluya interacciones con el usuario directamente.

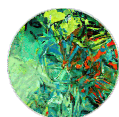
<pre>def suma(num_1, num_2):     resultado = num_1 + num_2     return resultado</pre>	<pre>&gt;&gt;&gt; suma(3,2) 5 &gt;&gt;&gt; suma(10,4) 14</pre>
---	--



### 3. Comentarios

- En este [enlace](#) puedes comprobar si has **llamado correctamente al repositorio** de trabajo de clase
  - Si lo has creado pero no aparece es porque posiblemente se llame de manera distinta.
- Descarga **Thonny** o **Visual Studio** en un ordenador y **prueba las soluciones**.
- Tienes más información en el [enlace](#) del grupo de 2º bachillerato.

AG, B	E1.1. Has leído correctamente las variables pero no lo has mostrado correctamente. E2.1. Deberías usar los operadores de comparación( > , < y ==) E4.2. Has repetido varias veces el mismo fragmento de código.
AS, SD	E1.1. E2.2. print para imprimir input para leer, tienes que asignarlo a una variable. E1.2. Los valores de las variables de cadenas de caracteres van entre comillas. E2.1. Si el número es igual a 0 entonces es 0. E4.2. def suma (x1, x2): res = x1 + x2 return res
BC, J	E1.1. E1.2. # lees por teclado texto = input() numero = int (input())  E1.2. Los valores de las variables de cadenas de caracteres van entre comillas.  E2.1. Deberías usar los operadores de comparación( > , < y ==)
CV, H	E1.1. Tienes que asignar a una variable lo que lees. # Ejemplo leer por teclado

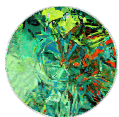


	<pre>texto = input() numero = int (input()) # Ejemplo de imprimir por consola print("El texto es: ", texto)</pre> <p>E1.2. (Int, float, string, boolean) son los tipos de datos.</p> <p>lenguaje = "Python" lenguaje es un identificador de la variable que tiene un tipo de dato String</p> <p>pi = 3.14 pi es un identificador de la variable que tiene un tipo de dato float</p>
DP, RL	<p>E1.1. print para imprimir input para leer</p> <p>E3.1. range(1,11) para mostrar desde 1 hasta 10</p>
GA, S	<p>E2.2. Necesitas dos números para poder comparar</p>
GO, L	<p>E1.1. print para imprimir input para leer</p>
GR, Á	<p>E1.2. valor=25 num=3.14 a="Python" E1.1. ·E1.2. # lees por teclado texto = input() numero = int (input())</p>
GR, S	<p>E1.1. print para imprimir input para leer</p> <p>E2.1. = es asignación == es para comparación</p> <p>E3.1. range(1,11) para mostrar desde 1 hasta 10</p>
GG, A	- Falta cuenta de GitHub

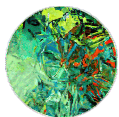


	<pre>E1.2. valor=25 num=3.14 a="Python" E1.1. ·E1.2. # lees por teclado texto = input() numero = int (input())</pre>
HD, M	<pre>E1.2. valor=25 num=3.14 a="Python" E2.1. ·E2.2. # lees por teclado num1 = int (input())</pre>
IC, A	<pre>E1.2. language="Python" numero=3.14 correcto=False</pre>
JO, P	<p>E3.2, E4.1: Realiza los ejercicios propuestos de clase</p>
LN, E	<pre>- Falta cuenta de GitHub E1.2. language="Python" numero=3.14 correcto=False E2.2. = es asignación == es para comparación E4.2 Las funciones devuelven el valor con return. def suma (x1, x2):     res = x1 + x2     return res</pre>
MP, M	<pre>- Falta cuenta de GitHub E1.1. print para imprimir input para leer E1.2. language="Python" numero=3.14</pre>





	<pre>correcto=False E2.1. = es asignación == es para comparación E.2.2 if : numero1 &gt; numero2 NO ES CORRECTO if(numero1&gt;numero&gt;2): ES LO CORRECTO</pre>
MC, V	<pre>E2.1. = es asignación == es para comparación</pre>
MM, JD	<pre>E1.1. "Capturas la edad correctamente; sin embargo, no utilizas adecuadamente la función print E1.2. language="Python" numero=3.14 correcto=False</pre>
NC, M	<pre>E4.2 Las funciones devuelven el valor con return. def suma (x1, x2):     res = x1 + x2     return res</pre>
RC, M	<pre>E1.1. print para imprimir input para leer E1.2. language="Python" numero=3.14 correcto=False E2.1. ·E2.2. # lees por teclado num1 = int (input()) texto = input() E2.1. = es asignación == es para comparación</pre>
RV, I	<pre>E1.1. print para imprimir input para leer E1.2.</pre>



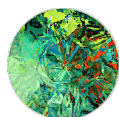
	<pre>language="Python" numero=3.14 correcto=False E2.1. ·E2.2. # lees por teclado num1 = int (input()) texto = input()</pre>
RV, M	<p>E2.1. = es asignación == es para comparación E4.2. Las funciones devuelven un valor con <b>return</b></p>
RB, D	<pre>E1.1. print para imprimir input para leer E1.2. language="Python" numero=3.14 correcto=False E2.1. ·E2.2. # lees por teclado num1 = int (input()) texto = input()</pre>
RS, L	<p>E2.1. E2.2.0. = es asignación == es para comparación</p>
SA, A	<pre>E1.1. print para imprimir input para leer E1.2. language="Python" numero=3.14 correcto=False E2.1. ·E2.2. # lees por teclado num1 = int (input()) texto = input()</pre>



2º Bachillerato

**TICO.2.A.1. Programación**

IES CAMAS Curso 2024/25



## 4. Resultados

Iniciales	E1.1	E1.2	E2.1.	E2.2.	E3.1.	E3.2.	E4.1.	E4.2.	Total	Iniciales
AG, B	0,5	1	0,25	0	0	0	0	0,25	2,0	AG, B
AS, SD	0,25	0,7	0,75	0,75	1	0	0	0,5	4,7	AS, SD
BC, J	0,25	0,7	0,25	0	0	0	0	0	1,2	BC, J
CV, H	0,5	1	0,75	0,5	0,5	0	0	0	3,8	CV, H
DP, RL	0,25	0	0	0,75	1	0	0	0	2,8	DP, RL
GA, S	1	0	0,25	0	0	0	0	0	1,3	GA, S
GO, L	0,5	0	0	0	0,5	0	0	0	1,0	GO, L
GR, Á	0	0	0,25	0	1	0	0	0	1,3	GR, Á
GR, S	0	0	0,5	0	0,5	0	0	1	2,0	GR, S
GG, A	1	0,3	1	1	0	0	0	0	4,3	GG, A
HD, M	0,5	0	0,5	0,75	1	0	0	0	3,5	HD, M
IC, A	0,25	0	1	1	1	0	0	0	4,3	IC, A
JO, P	1	1	1	1	1	0	0	1	7,0	JO, P
LN, E	0,5	0	1	0,5	1	0,25	0	0,5	4,5	LN, E
MP, M	0,5	0	0,75	0,5	0,75	0	0,25	1	4,3	MP, M
MC, V	1	1	0,75	1	1	1	1	1	9,8	MC, V
MM, JD	0,5	0	1	0	1	0	0	1	3,5	MM, JD
NC, M	1	1	1	1	1	0	0	0	6,0	NC, M
RC, M	0	0	0,25	0	0	0	0	0	0,3	RC, M
RV, I	0	0	1	0	1	0	0	0	2,0	RV, I
RV, M	1	0,7	0,75	1	1	0	0	0	5,4	RV, M
RB, D	0,5	0	0,5	0	0	0	0	1	2,0	RB, D
RS, L	1	1	0,75	0,75	1	0	0	0	5,3	RS, L
SA, A	0	0	0	0,25	0	0	0	0	0,5	SA, A
							Promedio:	3,43		