

# Programación con python

## Funciones

# Funciones

Una función es como un conjunto de instrucciones que te permite reutilizar un bloque de código en cualquier momento.

Sirve para organizar tu programa de manera lógica y dividirlo en partes más pequeñas y manejables.

# Funciones

En Python, una función se define con la palabra clave `def`, seguida del nombre de la función y, opcionalmente, de parámetros entre paréntesis.

```
def saludar():  
    print("¡Hola, mundo!")
```

Para ejecutarla, simplemente escribes su nombre seguido de paréntesis:

```
saludar()
```

# Argumentos y Parámetros

Los términos argumento y parámetro suelen usarse como sinónimos, pero hay una diferencia técnica:

Parámetro: es la variable que la función recibe (dentro de su definición).

Argumento: es el valor real que se pasa al llamar a la función.

# Argumentos y Parámetros

```
def saludar(nombre): # 'nombre' es el parámetro  
    print(f"Hola, {nombre}!")
```

```
saludar("Luis")      # "Luis" es el argumento
```

# Ventajas de usar funciones

- Evitar repetir código.
- Hacen el programa más fácil de leer y mantener.
- Permiten dividir tareas grandes en partes más simples.
- Pueden reutilizarse en diferentes programas.

# Valor de retorno (return)

Las funciones pueden devolver un resultado usando la palabra clave return.

Si no usas return, la función devuelve automáticamente None.

```
def sumar(a, b):  
    return a + b
```

```
resultado = sumar(5, 3)  
print(resultado) # 8
```

# Tipos de funciones según su uso

- Sin parámetros ni retorno:

```
def saludar():  
    print("Hola")
```

- Con parámetros

```
def saludar(nombre):  
    print(f"Hola, {nombre}")
```

- Con retorno de valor

```
def cuadrado(n):  
    return n * n
```



# Ejemplo

```
def calcular_area(base, altura):  
    area = (base * altura) / 2  
    return area
```

```
resultado = calcular_area(6, 4)  
print(f"El área es {resultado}")
```

# Parámetros con valores predeterminados

En Python, podemos definir valores por defecto para los parámetros de una función.

Esto significa que si el usuario no proporciona un valor, la función utilizará el valor definido por defecto.

```
def saludar(nombre="invitado"):
    print(f"Hola, {nombre}!")
```

```
saludar()      # Usa el valor por defecto → "Hola, invitado!"
```

```
saludar("Luis") # Usa el valor proporcionado → "Hola, Luis!"
```

# Argumentos por posición y por nombre

Cuando llamamos a una función, los argumentos pueden pasarse por posición o por nombre (keyword arguments).

- Argumentos por posición

Se asignan en el orden en que se definen los parámetros.

```
def mostrar_datos(nombre, edad):  
    print(f"Nombre: {nombre}, Edad: {edad}")
```

```
mostrar_datos("Ana", 20) # Por posición
```

# Argumentos por posición y por nombre

- Argumentos por nombre

Indicamos explícitamente qué valor corresponde a cada parámetro. Esto hace el código más claro y no depende del orden.

```
mostrar_datos(edad=25, nombre="Carlos") # Por nombre
```

# Combinación de ambos

Podemos mezclar argumentos por posición y por nombre, pero los de nombre deben ir al final:

```
def presentar(nombre, edad, ciudad="Desconocida"):
    print(f"{nombre}, {edad} años, vive en {ciudad}")
```

```
presentar("María", 30)                # Usa el valor por defecto de ciudad
presentar("Pedro", 28, "Guadalajara")  # Todos por posición
presentar("Sofía", edad=22, ciudad="CDMX") # Mezclado
```

# Actividad

- Crea una función llamada saludo() que:

No reciba parámetros.

Solo imprima un mensaje de bienvenida en pantalla.

Llama a la función varias veces desde el programa principal.

# Actividad

- Crea una función llamada presentar(nombre, edad) que reciba dos valores y muestre un mensaje como:

Hola, me llamo Luis y tengo 18 años.

Pide los datos con input() y pasalos a la función.

# Actividad

- Crea una función llamada sumar(a, b) que:

Reciba dos números.

Devuelva la suma.

Muestra el resultado fuera de la función con `print()`.



# Actividad

- Crea una función promedio(num1, num2, num3) que calcule el promedio de tres números y devuelva el resultado.

Luego imprime el promedio redondeado a 2 decimales.

# Actividad

- Crea una función saludar(nombre="invitado") que muestre:

Hola, [nombre]

Si el usuario no da un nombre, debe mostrar “Hola, invitado”.

Si sí lo da, debe mostrar el nombre que escribió.

# Actividad

- Crea una función datos(nombre, edad, ciudad) y haz tres llamadas:

Con todos los argumentos por posición.

Con todos por nombre.

Mezclando los dos tipos.

# Actividad

- Define una función `area_triangulo(base, altura)` que devuelva el área.  
Usa `return` y muestra el resultado en pantalla.

Fórmula:

$$\text{Área} = (\text{base} * \text{altura}) / 2$$

# Actividad

- Crea una función `celsius_a_fahrenheit(c)` que devuelva el valor convertido a Fahrenheit.

Usa la fórmula:

$$F = (C * 9/5) + 32$$

# Actividad

- Crea un programa con tres funciones:

`menu()` – muestra opciones.

`suma(a, b)` – devuelve la suma.

`resta(a, b)` – devuelve la resta.

Permite al usuario elegir qué operación desea realizar.

Usa `input()` y muestra el resultado final.