

## Práctico 2: Funciones en Python

### Objetivo:

Comprender y aplicar el uso de funciones en la programación, desarrollando algoritmos que implementen modularidad, reutilización de código y una organización estructurada para resolver problemas.

### Resultados de aprendizaje:

Fundamentos de las Funciones: A través de explicaciones teóricas y ejercicios prácticos, el estudiante desarrollará habilidades para definir, llamar y reutilizar funciones en Python.

Modularidad y Reutilización: El estudiante será capaz de descomponer problemas complejos en pequeñas unidades funcionales, mejorando la claridad, mantenimiento y reutilización del código.

Resolución de Problemas con Funciones: El estudiante aplicará funciones para resolver problemas computacionales que involucren cálculos matemáticos, manipulación de datos y flujos de control más complejos.

Buenas Prácticas: El estudiante aprenderá a estructurar el código usando funciones y a documentar adecuadamente cada una para facilitar su comprensión.

### Actividades:

1. Crear una función llamada `imprimir_hola_mundo` que imprima por pantalla el mensaje: "Hola Mundo!". Llamar a esta función desde el programa principal.
2. Crear una función llamada `saludar_usuario(nombre)` que reciba como parámetro un nombre y devuelva un saludo personalizado. Por ejemplo, si se llama con `saludar_usuario("Marcos")`, deberá devolver: "Hola Marcos!". Llamar a esta función desde el programa principal solicitando el nombre al usuario.
3. Crear una función llamada `informacion_personal(nombre, apellido, edad, residencia)` que reciba cuatro parámetros e imprima: "Soy [nombre] [apellido], tengo [edad] años y vivo en [residencia]". Pedir los datos al usuario y llamar a esta función con los valores ingresados.

4. Crear dos funciones: `calcular_area_circulo(radio)` que reciba el radio como parámetro y devuelva el área del círculo. `calcular_perimetro_circulo(radio)` que reciba el radio como parámetro y devuelva el perímetro del círculo. Solicitar el radio al usuario y llamar ambas funciones para mostrar los resultados.
5. Crear una función llamada `segundos_a_horas(segundos)` que reciba una cantidad de segundos como parámetro y devuelva la cantidad de horas correspondientes. Solicitar al usuario los segundos y mostrar el resultado usando esta función.
6. Crear una función llamada `tabla_multiplicar(numero)` que reciba un número como parámetro y imprima la tabla de multiplicar de ese número del 1 al 10. Pedir al usuario el número y llamar a la función.
7. Crear una función llamada `operaciones_basicas(a, b)` que reciba dos números como parámetros y devuelva una tupla con el resultado de sumarlos, restarlos, multiplicarlos y dividirlos. Mostrar los resultados de forma clara.
8. Crear una función llamada `calcular_imc(peso, altura)` que reciba el peso en kilogramos y la altura en metros, y devuelva el índice de masa corporal (IMC). Solicitar al usuario los datos y llamar a la función para mostrar el resultado con dos decimales.
9. Crear una función llamada `celsius_a_fahrenheit(celsius)` que reciba una temperatura en grados Celsius y devuelva su equivalente en Fahrenheit. Pedir al usuario la temperatura en Celsius y mostrar el resultado usando la función.
10. Crear una función llamada `calcular_promedio(a, b, c)` que reciba tres números como parámetros y devuelva el promedio de ellos. Solicitar los números al usuario y mostrar el resultado usando esta función.

### **Consejo:**

Antes de empezar, analiza cada problema y piensa cómo dividirlo en pasos más pequeños utilizando funciones.

Al terminar, prueba cada función con diferentes entradas para verificar que funciona correctamente.