

DÍA 5 – LÓGICA APLICADA CON FUNCIONES

Objetivo

- Consolidar conocimientos previos (listas, condicionales, funciones)
- Resolver problemas reales usando funciones
- Aprender validaciones prácticas con input del usuario
- Aplicar lógica: IMC (Índice de Masa Corporal) y cálculo de año de nacimiento

EJEMPLO 1 – CALCULAR EL IMC

¿Qué **es** el IMC?

Una fórmula para saber **si** tu peso **es** saludable:

$$\text{IMC} = \frac{\text{peso}}{\text{altura}^2}$$

```
def calcular_imc(peso, altura):  
    return peso / (altura ** 2)
```

```
def clasificar_imc(imc):  
    if imc < 18.5:  
        return "Bajo peso"  
    elif imc < 25:  
        return "Peso normal"  
    elif imc < 30:  
        return "Sobrepeso"  
    else:  
        return "Obesidad"  
  
peso = float(input("¿Cuál es tu peso (kg)? "))  
altura = float(input("¿Cuál es tu altura (m)? "))  
  
imc = calcular_imc(peso, altura)  
categoria = clasificar_imc(imc)  
  
print(f"Tu IMC es {imc:.2f} y estás en la categoría: {categoria}")
```

Analogía:

- calcular_imc() → calculadora
- clasificar_imc() → etiqueta que dice "estás OK" o "cuidado"
- input() → preguntas que haces al paciente

EJEMPLO 2 – AÑO DE NACIMIENTO

```
def calcular_nacimiento(edad):  
    from datetime import date  
    año_actual = date.today().year  
    return año_actual - edad  
  
nombre = input("¿Cómo te llamas? ")  
edad = int(input("¿Cuántos años tienes? "))  
  
año = calcular_nacimiento(edad)  
  
print(f"{nombre}, naciste en el año {año}")
```

Validaciones opcionales

- Edad no puede **ser** negativa
- Altura **y** peso deben **ser** mayores que 0

```
if peso <= 0 or altura <= 0:  
    print("    Datos inválidos")  
else:  
    imc = calcular_imc(peso, altura)
```

DÍA 5 – LÓGICA APLICADA CON FUNCIONES

Objetivo del día

- Aplicar funciones a problemas reales
- Usar condicionales y entrada de datos para personalizar respuestas
- Calcular el ****IMC (Índice de Masa Corporal)**** y el ****año de nacimiento**** de una persona
- Validar correctamente la entrada de datos

TABLA RESUMEN – PUNTOS CLAVE DEL DÍA 5

| Concepto | Descripción breve | Detalles clave |
|----------------------------|--|--|
| def | Palabra clave para crear una función | Sigue con el nombre y paréntesis: def nombre(parámetros): |
| return | Devuelve un resultado desde una función | Permite guardar o usar ese valor más adelante |
| IMC (Índice Masa Corporal) | Fórmula: peso / altura**2 | Sirve para clasificar salud corporal en bajo peso, normal, sobrepeso, obesidad |
| Función calcular_imc() | Recibe peso y altura, devuelve el resultado del IMC | Debe incluir altura**2 en el denominador |
| Función clasificar_imc() | Recibe el resultado del IMC y devuelve texto (categoría) | Usa condicionales: if, elif, else |
| input() | Pide datos al usuario | Conviene convertir con int() o float() según sea edad, peso, etc. |
| from datetime import date | Sirve para obtener el año actual automáticamente | date.today().year devuelve el año actual, útil para cálculo de año de nacimiento |
| Validación de datos | Comprobar si los datos son válidos antes de operar | if peso <= 0 o if edad < 0 para evitar errores o resultados sin sentido |
| Lógica aplicada | Resolver problemas reales con funciones + condiciones + entrada de usuario | Ejemplo: IMC, nacimiento, votación, clasificaciones personalizadas |
| Pensamiento estructurado | Dividir cada parte del problema en funciones | 1 tarea → 1 función clara. Reutilizable, legible y fácil de probar |