

## Práctica 5

### Instrucciones:

Resuelva **individualmente** los siguientes ejercicios. No se reciben trabajos fuera de la fecha establecida en la plataforma *Canvas*:

- Entregable: Archivo.pdf
  - El nombre del archivo será “P, #Práctica, Primer Nombre, Primer Apellido”.
  - Ejemplo: “P1LuisAlvarado.pdf”

La calificación de esta práctica estará distribuida equitativamente entre los ejercicios. Agrega los comentarios necesarios al archivo.

Escribe el código e invoca una función en Python para cada uno de los siguientes casos. Antes de empezar, considera si la función necesita recibir parámetros (cuántos y cuáles, eso lo decides **TÚ**), si debe mostrar algo en pantalla y si debe devolver algún valor. Ninguna función deberá pedir los datos directamente al usuario (no inputs).

### Ejercicios:

1. A partir de tres cadenas, devolver una sola cadena que las contenga separadas por guiones (-).
2. Diseña una función que reciba tres números y devuelva el menor (no puedes usar `min()`).
3. A partir de un número entero  $n$  entre 0 y 999, devuelve una lista que contenga las unidades, decenas y centenas.
4. Diseña una función que reciba tres enteros, verifique si son válidos (positivos y menores a 1000) y luego los clasifique en base a las siguientes reglas:
  - Si todos son múltiplos de 3, devuelve "Múltiplos de 3".
  - Si dos de ellos son pares y el tercero es impar, devuelve "Dos pares y un impar".
  - Si los tres son números primos, devuelve "Todos primos".
  - Si no cumple ninguna de las condiciones anteriores, devuelve "Sin clasificación".

5. Diseña una función que reciba una lista de puntuaciones en porcentaje y devuelva una letra según las siguientes condiciones:
- Mayor o igual a 90: "A"
  - Entre 80 y 89: "B"
  - Entre 70 y 79: "C"
  - Entre 60 y 69: "D"
  - Menor de 60: "F"

La función debe validar que los porcentajes estén entre 0 y 100, y mostrar un error si no cumplen.

6. Diseña una función que reciba el ingreso anual de una persona y calcule el impuesto según el siguiente esquema progresivo:
- Primeros \$10,000 libres de impuestos.
  - Siguiendo \$30,000 al 10%.
  - Todo lo que exceda \$40,000 al 20%.

La función debe devolver el monto total de impuestos a pagar.

7. Diseña una función que reciba una lista de números enteros y devuelva una nueva lista con una clasificación compleja. Para cada número:
- Si es par y mayor que 100, devuelve "Par grande".
  - Si es impar y menor que 50, devuelve "Impar pequeño".
  - Si es divisible por 7 y negativo, devuelve "Negativo múltiplo de 7".
  - Para cualquier otro caso, devuelve "Sin clasificación".
8. Diseña una función que reciba una contraseña y valide si cumple las siguientes condiciones:
- Tiene al menos 8 caracteres.
  - Contiene al menos una mayúscula, una minúscula, un número y un carácter especial (como @, #, !, etc.).
  - No contiene espacios.

Si la contraseña cumple con todas las condiciones, devuelve "Contraseña válida"; de lo contrario, devuelve "Contraseña inválida".

9. Diseña una función que reciba una lista de números y determine si sigue algún patrón específico:
- Si todos los números son pares y aumentan, devuelve "Secuencia ascendente par".
  - Si todos los números son impares y disminuyen, devuelve "Secuencia descendente impar".
  - Si la lista alterna entre números pares e impares devuelve "Secuencia alternada".
  - Si no sigue ningún patrón, devuelve "Sin patrón".
10. Diseña una función que simule una ronda de "piedra, papel o tijera", pero agregando dos opciones adicionales: "lagarto" y "Spock". Las reglas son las siguientes:
- Tijera corta papel.
  - Papel cubre piedra.
  - Piedra aplasta lagarto.
  - Lagarto envenena Spock.
  - Spock rompe tijera.
  - Tijera decapita lagarto.
  - Lagarto devora papel.
  - Papel desautoriza Spock.
  - Spock vaporiza piedra.
  - Piedra rompe tijera.

La función debe recibir dos entradas (jugador 1 y jugador 2) y determinar quién gana o si es un empate.