

## GUÍA DE EJERCICIOS

### LISTAS

#### Creación de Listas

1. Crea una lista llamada `numeros` que contenga los números del 1 al 10.
2. Usa `list()` y `range()` para crear una lista llamada `pares` que contenga los números pares del 2 al 20.
3. Crea una lista vacía llamada `vacía` y agrégle los números 5, 10, y 15 utilizando el método `append()`.

#### Acceso a Elementos y Rebanado

4. Dada la lista `colores = ["rojo", "verde", "azul", "amarillo", "negro"]`, accede y muestra el segundo y el cuarto elemento.
5. Usa el rebanado para obtener una sublista con los tres primeros elementos de la lista `colores`.
6. Invierte el orden de los elementos en la lista `colores` sin usar el método `reverse()`.

#### Operadores con Listas

7. Crea una lista llamada `animales` que contenga los elementos `["gato", "perro", "pájaro"]`. Luego, agrega otra lista `["pez", "hamster"]` utilizando el operador `+`.
8. Repite la lista `animales` dos veces utilizando el operador `*` y muestra el resultado.
9. Verifica si el elemento `"gato"` está en la lista `animales` usando el operador `in`.

#### Funciones de Listas

10. Dada la lista `numeros = [5, 10, 15, 20, 25]`, encuentra:
  - La longitud de la lista usando `len()`.
  - La suma de los elementos usando `sum()`.
  - El valor mínimo y máximo de la lista usando `min()` y `max()`.

#### Métodos de Listas

11. Dada la lista `frutas = ["manzana", "pera", "cereza", "manzana"]`:
  - Usa el método `count()` para contar cuántas veces aparece `"manzana"`.
  - Usa el método `index()` para encontrar el índice de la primera aparición de `"pera"`.
  - Usa el método `remove()` para eliminar `"cereza"` de la lista.
  - Ordena la lista `frutas` alfabéticamente usando `sort()`.
  - Inserta el elemento `"kiwi"` en la segunda posición de la lista `frutas`.

#### Recorrido de Listas

12. Dada la lista `numeros = [1, 2, 3, 4, 5]`, recorre la lista utilizando un bucle `for` e imprime cada número.
13. Repite el ejercicio anterior utilizando un bucle `while`.
14. Modifica el bucle `for` para que solo imprima los números pares de la lista.

## Copiado de Listas

15. Crea una lista llamada original = [10, 20, 30]. Luego, haz una copia de esta lista llamada copia utilizando el método copy().
16. Modifica la lista copia agregando el número 40. Imprime ambas listas para demostrar que son listas independientes.

## Listas de Listas (Matrices)

17. Crea una lista de listas (matriz) llamada matriz, que represente la siguiente tabla:
18. Accede al número 5 en la matriz y muéstralo.
19. Usa un ciclo anidado for para recorrer la matriz y mostrar cada número.

## Ciclos Anidados

20. Usando la lista de listas creada en el ejercicio anterior, recorre todos los elementos e imprime solo aquellos que sean mayores que 5.
21. Modifica el ciclo para que sume todos los elementos de la matriz y muestre el resultado final.

## Uso de random.choice()

22. Crea una lista llamada nombres que contenga los nombres "Ana", "Carlos", "Pedro", "María", "Lucía". Usa la función random.choice() para seleccionar y mostrar un nombre al azar de la lista.

## Desafío

23. Desarrolla un algoritmo en Python que permita simular un juego de lotería. El programa debe pedir al usuario que ingrese 6 números entre 1 y 41, verificando que los números no se repitan, y el monto en dinero que apostará considerando que la apuesta mínima es \$1000. El programa realizará el sorteo simulando una tómbola generando 6 números distintos al azar entre 1 y 41 y mostrará al usuario los resultados del sorteo considerando lo siguiente:
  - a) Debe mostrar números seleccionados por el usuario ordenados de menor a mayor y el monto apostado.
  - b) Debe mostrar los números sorteados ordenados de menor a mayor.
  - c) Debe mostrar la cantidad de aciertos logrados por el usuario.
  - d) Si corresponde, debe mostrar el premio ganado de acuerdo a la siguiente tabla:

Cantidad de Aciertos	Premio
1	1 vez lo apostado
2	5 veces lo apostado
3	10 veces lo apostado
4	500 veces lo apostado
5	1000 veces lo apostado
6	10.000 veces lo apostado