

Práctico 5: Listas

Objetivo:

Desarrollar la comprensión y la capacidad de manipular listas en Python mediante la aplicación de conceptos fundamentales como la indexación, la modificación de elementos, el uso de métodos integrados y el manejo de listas anidadas.

Resultados de aprendizaje:

1. Reconocer y aplicar correctamente la indexación y el slicing para acceder a elementos individuales o subconjuntos dentro de una lista.
2. Utilizar los métodos básicos de listas para crear, modificar y gestionar estructuras de datos simples.
3. Modificar listas mediante la actualización de valores y el manejo de listas anidadas, comprendiendo cómo acceder a datos en estructuras más complejas.

Actividades

- 1) **Crear una lista con los números del 1 al 100 que sean múltiplos de 4. Utilizar la función range.**

```
lista_multiplos_de_4 = list(range(4, 101, 4))  
print(lista_multiplos_de_4)
```

- 2) **Crear una lista con cinco elementos (colocar los elementos que más te gusten) y mostrar el penúltimo. ¡Puedes hacerlo como se muestra en los videos o bien investigar cómo funciona el indexing con números negativos!**

```
lista = list(range(0, 5))  
print(lista[4])
```

- 3) **Crear una lista vacía, agregar tres palabras con append e imprimir la lista resultante por pantalla. Pista: para crear una lista vacía debes colocar los corchetes sin nada en su interior. Por ejemplo:**

```
lista = list()  
lista.append('elemento1')  
lista.append('elemento2')  
lista.append('elemento3')  
print(lista)
```

- 4) **Reemplazar el segundo y último valor de la lista “animales” con las palabras “loro” y “oso”, respectivamente. Imprimir la lista resultante por pantalla. ¡Puedes hacerlo como se muestra en los videos o bien investigar cómo funciona el indexing con números negativos!**

```
animales = ["perro", "gato", "conejo", "pez"]
animales [1] = 'loro'
animales [2] = 'oso'
print(animales)
```

- 5) Analizar el siguiente programa y explicar con tus palabras qué es lo que realiza.

1

```
numeros = [8, 15, 3, 22, 7]
numeros.remove(max(numeros))
print(numeros)
```

En el primer renglón se crea una lista que contiene números, en el segundo renglón, la función max busca el valor máximo dentro de la lista números, y números.remove remueve ese numero máximo anteriormente llamado.

El 3er renglón nos devuelve por pantalla lo que contiene la variable, que en este caso será una lista con los números [8, 15, 3, 7]

- 6) Crear una lista con números del 10 al 30 (incluido), haciendo saltos de 5 en 5 y mostrar por pantalla los dos primeros.

```
lista = list(range(10, 31, 5))
print(lista[:2])
```

- 7) Reemplazar los dos valores centrales (índices 1 y 2) de la lista "autos" por dos nuevos valores cualesquiera.

```
autos = ["sedan", "polo", "suran", "gol"]
autos [1] = 'loro'
autos [2] = 'oro'
print(autos)
```

- 8) Crear una lista vacía llamada "dobles" y agregar el doble de 5, 10 y 15 usando append directamente. Imprimir la lista resultante por pantalla.

```
dobles = []
dobles.append(5 * 2)
dobles.append(10 * 2)
dobles.append(15 * 2)
```

9) Dada la lista “compras”, cuyos elementos representan los productos comprados por diferentes clientes:

```
compras = [["pan", "leche"], ["arroz", "fideos", "salsa"],  
["agua"]]
```

- Agregar "jugo" a la lista del tercer cliente usando append.
- Reemplazar "fideos" por "tallarines" en la lista del segundo cliente.
- Eliminar "pan" de la lista del primer cliente.
- Imprimir la lista resultante por pantalla

```
compras = [["pan", "leche"], ["arroz", "fideos", "salsa"],["agua"]]  
compras[2].append("jugo")  
indice_fideos = compras[1].index("fideos")  
compras[1][indice_fideos] = "tallarines"  
compras[0].remove("pan")  
print(compras)
```

10) Elaborar una lista anidada llamada “lista_anidada” que contenga los siguientes elementos:

- Posición lista_anidada[0]: 15
- Posición lista_anidada[1]: True
- Posición lista_anidada[2][0]: 25.5
- Posición lista_anidada[2][1]: 57.9
- Posición lista_anidada[2][2]: 30.6
- Posición lista_anidada[3]: False

```
lista_anidada = [15,True, [25.5, 57.9, 30.6],False]  
print(lista_anidada)
```