



Programación Avanzada 2025

Lab. 1.3. Listas en Python, Parte I

Agosto 23, 2025

Cree una carpeta (folder) en el disco D, nómbrela con su apellido paterno seguido de su código. Ejemplo: LOPEZ12345

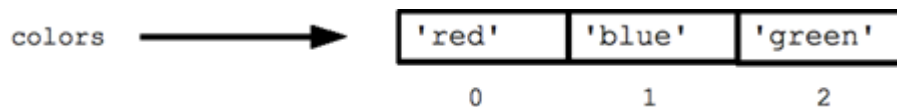
Una lista es una secuencia de valores llamados items o elementos. Un ítem o elemento puede ser de cualquier tipo. A diferencia de los strings, las listas si permiten modificar su contenido, es decir, son mutables.

En Python, una lista se crea colocando todos sus elementos entre corchetes separados por comas. Ejemplo:

```
number_list = [4, 4, 6, 7, 2, 9, 10, 15]
mixed_list = ['dog', 87.23, 65, [9, 1, 8, 1]]
empty_list = []
```

Al igual que en un string, las listas mantienen sus elementos indexados, el primer elemento lleva el índice 0.

```
colors = ['red', 'blue', 'green']
```



```
print(colors[0])
>>> red

print(colors[2])
>>> green

print(len(colors))
>>> 3
```

Modificando valores de una lista

Para modificar un elemento de una lista, basta apuntar al elemento mediante su índice y asignarle un nuevo valor, ejemplo:

```
num = [1,2,3,4,5]
num[3] = 0
print(num)
>>> [1,2,3,0,5]
```

Recorrido de una lista

Una lista se recorre de la misma forma que recorre una cadena de caracteres (string), es decir con la ayuda de un bucle.

Para ejercicios del 1 al 3, verificar la salida con la ayuda del interpretador de Python

1. Dada la lista `lst=[30, 1, 2, 1, 0]`, ¿qué lista se genera al aplicar cada una de las siguientes instrucciones?. Asuma que cada línea de código es independiente.
 - a) `lst.append(40)`
 - b) `lst.insert(1, 43)`
 - c) `lst.extend([1, 43])`
 - d) `lst.remove(1)`
 - e) `lst.pop(1)`
 - f) `lst.pop()`
 - g) `lst.sort()`
 - h) `lst.reverse()`
 - i) `random.shuffle(lst)`
2. dada la lista `lst=[30, 1, 2, 1, 0]`, ¿cuál es el valor que retorna las siguientes instrucciones?
 - a) `lst.index(1)`
 - b) `lst.count(1)`
 - c) `len(lst)`
 - d) `max(lst)`
 - e) `min(lst)`
 - f) `sum(lst)`
3. Dada la lista `list1=[30, 1, 2, 1, 0]` and `list2=[1, 21, 13]`, ¿qué valor devuelve cada una de las siguientes instrucciones?
 - a) `list1 + list2`
 - b) `2 * list2`
 - c) `list2 * 2`
 - d) `list1[1 : 3]`
 - e) `list1[3]`

Para los siguientes ejercicios, elabore un programa en Python. Verifique los programas ejecutando y probando con distintas entradas.

1. Guardando valores en una lista.
 - a) Defina una lista vacía A. Leer del teclado N números enteros y colocarlos en la lista A. Adicione uno a uno los elementos de la lista con la ayuda del método `append()`.
 - b) Inicialice la lista B de tamaño N con valores iniciales iguales a `None`. Leer del teclado N números enteros y colocarlos en la lista B. Coloque uno a uno los números ingresados en cada posición de la lista, use los índices de los elementos (no usar `append()`).
 - c) Guardar en una nueva lista C los valores de A que estén ubicados entre la ubicación del máximo y la ubicación del mínimo.
 - d) Guardar en una nueva lista D los valores de B que estén sobre el promedio.
2. Crear dos listas A y B de `m` elementos numéricos cada uno.
 - a) Los elementos de la lista A deben de ser ingresados del teclado
 - b) Los elementos de B deben de ser generados aleatoriamente del intervalo `(-30,30)`
 - c) Guarde las listas en un archivo, cada lista en una fila.
 - d) Muestre el contenido del archivo
 - e) Sumar las listas A y B elemento a elemento y guardarlos en otra lista C.
 - f) Agregue la lista C al archivo donde se encuentran las listas A y B.

3. En el archivo del ejercicio anterior, modificar el contenido de la lista B de la siguiente manera, colocar todos los números negativos en las primeras posiciones y luego el resto de los números. Mostrar el contenido del archivo modificado.
4. Leer con la ayuda de Python el archivo que se encuentra en la siguiente dirección:
<https://www.gutenberg.org/files/2000/2000-0.txt>

Realizar lo siguiente:

- a) Determinar la cantidad de palabras que contiene el archivo
- b) Mostrar las 10 primeras líneas
- c) Mostrar las palabras que se encuentran en el archivo solo una vez.
- d) Contar las palabras que tengan longitud mayor a 2 y que su primer y último carácter sean iguales.

Guarde todos vuestros programas en una carpeta con el nombre su **Apellido** paterno seguido de vuestro **DNI**, luego comprima esta carpeta. Envíe este archivo a:
Katherine Navarro katherine.navarro@upch.pe especificando como asunto **Lab1.3**.