



Universidad Autónoma de Baja California

Facultad de Ingeniería, Arquitectura y Diseño

Ingeniero en Computación

Asignatura:

Lenguaje de Programación Python

Actividad 7:

Listas

Brayan Arturo Rocha Meneses

Matricula:

371049

Ensenada Baja California 4 de Octubre del 2023

Introducción:

Las listas son una estructura de datos fundamental en el lenguaje de programación Python. Son utilizadas para almacenar y gestionar colecciones de elementos de manera ordenada y mutable. En Python, las listas son muy versátiles y pueden contener una variedad de tipos de datos, como números, cadenas de texto, booleanos e incluso otras listas.

Una de las características más destacadas de las listas en Python es su capacidad para cambiar dinámicamente su tamaño, lo que significa que puedes agregar, eliminar o modificar elementos fácilmente. Además, las listas se definen utilizando corchetes `[]` y los elementos individuales se separan por comas.

Competencia:

Esta competencia está diseñada para ayudar a fortalecer las habilidades en el manejo de listas y para que puedas aplicar los conceptos básicos que has aprendido

Fundamentos:

Introducción a listas En Python, una lista es un tipo de estructura de datos que se utiliza para almacenar una colección de elementos en un orden determinado. Las listas son muy versátiles se utilizan comúnmente en la programación de Python para manejar datos de manera eficiente.

¿Qué son y uso de las listas?

Las listas en Python son colecciones ordenadas de elementos, que se pueden modificar. Las listas son útiles para almacenar colecciones de datos relacionados, como una lista de números o una lista de nombres. Las listas son similares a los arrays en otros lenguajes de programación, pero tienen algunas características adicionales que los hacen más útiles.

Sintaxis En Python,

las listas se definen utilizando corchetes [], y los elementos de la lista están separados por comas ,. Por ejemplo, para crear una lista de números enteros, se podría usar la siguiente sintaxis:
`mi_lista = [1, 2, 3, 4, 5]`
Crear una lista vacía Para crear una lista vacía en Python, simplemente se puede definir una lista sin elementos utilizando corchetes:
`# Crear una lista vacía
mi_lista = []`

Python Agregar e imprimir listas

Para agregar elementos a una lista en Python, se puede utilizar el método `append()`. Este método agrega un elemento al final de la lista. Por ejemplo:
`# Creando una lista vacía
mi_lista = []
Agregando elementos a la lista
mi_lista.append("manzana")
mi_lista.append("banana")
mi_lista.append("cereza")
Imprimiendo la lista
print(mi_lista)`
Para imprimir un elemento específico de la lista, se puede utilizar la sintaxis `mi_lista[indice]`, donde índice es el número de índice del elemento que se desea imprimir.

Los índices en Python comienzan en 0. Por ejemplo, para imprimir el segundo elemento de la lista, se podría usar la siguiente sintaxis:
`# Imprimiendo el segundo elemento de la lista
print(mi_lista[1])`
Métodos de las listas en Python, las listas tienen varios métodos útiles que se pueden utilizar para manipular y trabajar con ellas. Algunos de los métodos más comunes incluyen:

- `append()`: Agrega un elemento al final de la lista.
- `insert()`: Agrega un elemento en una posición específica en la lista.
- `remove()`: Elimina el primer elemento de la lista que coincide con el valor dado.
- `pop()`: Elimina y devuelve el elemento en la posición especificada de la lista.
- `sort()`: Ordena los elementos de la lista en orden ascendente.
- `reverse()`: Invierte el orden de los elementos en la lista.

Procedimiento:

ACTIVIDAD 7 ejercicios

1.- Función que utilice una lista con los nombres de tus mascotas, o artistas favoritos, (minimo 5, maximo 10) imprimir las cadenas y la cantidad de caracteres de cada cadena.

EJEMPLO SALIDA:

```
[0] SCARLETT --> 8 CARACTERES
[1] GATINA --> 6 CARACTERES
[2] PELUSA --> 6 CARACTERES
[3] PELUSINA --> 8 CARACTERES
[4] MILO --> 4 CARACTERES
[5] KHALIFA --> 7 CARACTERES
```

2.- Programa que utilice 2 funciones,

- Función que genere y regrese una lista con 10 números aleatorios entre el 30 y 50 (no repetidos).
- Función que reciba una lista , Imprimir la lista (Indice y Valor)

Mandar llamar la función e imprimir toda la lista.

3.- Escribe una función que reciba dos listas de números del mismo tamaño y calcule la **suma de los elementos correspondientes de cada lista**. (regresar una nueva lista) . Luego, muestra una lista con los resultados de cada suma. Nota: si las listas no son del mismo tamaño mandar msge y utilizar el tamaño de la lista mas pequeña)

4.- Escribe una función llamada **eliminar_duplicados** que reciba una lista como parámetro y elimine los elementos duplicados. El resultado debe ser una nueva lista sin duplicados. Pide al usuario que ingrese una lista y luego llama a la función **eliminar_duplicados** para mostrar la lista sin duplicados. (Solo Numeros enteros, validar)

5.- Escribe una función que calcule la **media y la mediana** de una lista de números enteros.

La media es la suma de todos los elementos y dividido entre la cantidad de elementos

La mediana es el valor que queda en la mitad de la lista cuando se ordena de forma ascendente. Si la lista tiene un número par de elementos, la mediana se calcula como el promedio de los dos valores centrales.

Puedes utilizar el método `sort()` para ordenar la lista y luego calcular la mediana según el tamaño de la lista.

Conclusiones:

Las listas en Python, exploraremos cómo crear listas, acceder a sus elementos, realizar operaciones comunes como añadir y eliminar elementos, y comprender su importancia en la programación de Python. Las listas son una herramienta esencial en la caja de herramientas de todo programador de Python y son ampliamente utilizadas en una variedad de aplicaciones.

Anexo:

https://github.com/BrayanArM98/Python/blob/main/RMBA_PY_ACT7.ipynb