

LENGUAJES DE PROGRAMACIÓN

PYTHON





Listas y tuplas.

Python es un lenguaje de programación versátil y poderoso que ofrece una amplia gama de estructuras de datos para almacenar y manipular información. Dos de las estructuras de datos más fundamentales en Python son las listas y las tuplas. Tanto las listas como las tuplas son secuencias ordenadas de elementos, pero difieren en términos de mutabilidad y sintaxis. En esta lección, exploraremos en detalle las listas y las tuplas, sus características y cómo utilizarlas eficientemente en Python.





Listas

Una lista en Python es una colección ordenada de elementos separados por comas y encerrados entre corchetes. Cada elemento de una lista puede ser de cualquier tipo de datos válido, como números, cadenas de texto, booleanos e incluso otras listas. La principal característica de las listas es que son mutables, lo que significa que se pueden modificar después de su creación. Veamos un ejemplo:

```
# Crear una lista
mi_lista = [1, 2, 3, 4, 5]

# Acceder a los elementos de la lista
primer_elemento = mi_lista[0]
segundo_elemento = mi_lista[1]

# Modificar un elemento de la lista
mi_lista[2] = 10

# Añadir elementos a la lista
mi_lista.append(6)

# Eliminar elementos de la lista
del mi_lista[3]

# Longitud de la lista
longitud_lista = len(mi_lista)
```



En el ejemplo anterior, creamos una lista llamada `mi_lista` que contiene los números del 1 al 5. Luego, accedemos a los elementos de la lista utilizando el índice entre corchetes (los índices en Python comienzan desde 0). También podemos modificar un elemento de la lista asignando un nuevo valor a través del índice correspondiente. Además, podemos agregar elementos a la lista utilizando el método `append()`, y eliminar elementos utilizando la declaración `del`. La función `len()` nos permite obtener la longitud de la lista.

Tuplas

A diferencia de las listas, las tuplas son estructuras de datos inmutables en Python. Esto significa que una vez creada una tupla, no se pueden modificar sus elementos. Las tuplas se crean separando los elementos por comas y encerrándolos entre paréntesis. Veamos un ejemplo:

```
# Crear una tupla
mi_tupla = (1, 2, 3, 4, 5)

# Acceder a los elementos de la tupla
primer_elemento = mi_tupla[0]
segundo_elemento = mi_tupla[1]

# No se puede modificar una tupla existente
# mi_tupla[2] = 10 <-- Esto generará un error

# Longitud de la tupla
longitud_tupla = len(mi_tupla)
```



En el ejemplo anterior, creamos una tupla llamada `mi_tupla` que contiene los números del 1 al 5. Al igual que con las listas, podemos acceder a los elementos de la tupla utilizando el índice entre corchetes. Sin embargo, si intentamos modificar un elemento de la tupla, Python generará un error debido a su naturaleza inmutable. La función `len()` también se puede utilizar para obtener la longitud de una tupla.

Comparación entre listas y tuplas

La elección entre utilizar una lista o una tupla dependerá de los requisitos específicos de tu programa. A continuación, se presentan algunas consideraciones para tener en cuenta:

Si necesitas una estructura de datos mutable que te permita agregar, eliminar o modificar elementos, debes utilizar una lista.

Si necesitas una estructura de datos inmutable que garantice la integridad de los elementos y no permita modificaciones accidentales, debes utilizar una tupla.

Las listas son más flexibles y versátiles, pero pueden ocupar más espacio en memoria que las tuplas.

Las tuplas son más eficientes en términos de rendimiento y pueden ser utilizadas como claves en diccionarios.



Las listas y las tuplas son estructuras de datos fundamentales en Python que te permiten almacenar y manipular información de manera ordenada. Las listas son mutables, lo que significa que se pueden modificar, agregar o eliminar elementos, mientras que las tuplas son inmutables y no se pueden modificar después de su creación. La elección entre listas y tuplas dependerá de las necesidades y requisitos específicos de tu programa.

