Que entiende por indexación de los elementos en una colección de datos.

Una colección representa un grupo de objetos, conocidos como sus elementos. Las colecciones permiten elementos duplicados y los mismos no están ordenados. Los elementos de una colección no son accedidos a través de un índice. el contrario su única forma de recuperación es uno a uno, secuencialmente.

Como eliminar

Usando clear() método

La solución estándar para eliminar todos los elementos de una lista es usar el clear() método, que eficientemente hace que la lista esté vacía. Tenga en cuenta que solo funciona en listas mutables y lanza UnsupportedOperationException por lista no modificable.

El siguiente es el código fuente del clear() método para la implementación 'ArrayList' del List interfaz. Establece el búfer de array en el que se almacenan los elementos de la Lista en nulo. Para implementaciones de listas basadas en árboles, simplemente establece la raíz en nulo.

2. Usando removeAll() método

Una idea alternativa es llamar a la removeAll() método, que elimina todos los elementos de una lista que están presentes en la colección especificada

. Usando remove() método

A partir de Java 8, podemos obtener un flujo de elementos en la lista y llamar al remove() método para cada elemento.

Como agregar

Esto se demuestra a continuación. Tenga en cuenta que hemos utilizado una copia de la lista para evitar java.util.ConcurrentModificationException, ya que Java no permite la modificación simultánea en el List instancia mientras se itera sobre ella

El punto es muy importante porque append() es un método. Cuando llamamos a un método, escribimos el nombre de la lista seguida de un punto para indicar que queremos modificar esa lista en particular.

El primer elemento de la sintaxis (la lista) usualmente es una variable.

Como puedes ver, para llamar al método append() también debemos especificar un argumento, el elemento que deseamos agregar a la lista. Este elemento puede ser de cualquier tipo de dato:

Un entero (int).

Una cadena de caracteres (str).

Un número de coma flotante (float).

Valores Booleanos.

Una lista (list).

Una tupla.

Un diccionario (dict).

Una instancia de una clase definida en el programa.

Básicamente, cualquier valor que podemos crear y usar en Python puede ser agregado a una lista con el método append().

Ejemplo

Este es un ejemplo de cómo puedes llamar al método append():

Primero, se define la lista y se asigna a una variable.

Luego, usando esta variable llamamos al método append(), pasando el elemento que queremos agregar como argumento. En este caso, el elemento es la cadena de caracteres "Si".

Recorrer

No sé si podréis ayudarme, tengo una lista de listas y me gustaría acceder a todos los elementos de las sublistas pero que continuaran estando divididos. Mi lista podría ser algo parecido a esto:

lista=[["PATATA","GARBABANZOS"],["MELOCOTONES","PERAS"],["LECHE","CHOCOLATE","GALLETAS"]]

y yo por ejemplo, quiero pasar todas las cadenas a minúsculas, pero que sigan estando dentro de sus listas. Solo consigo pasar los elementos de una lista, no de todas.

compra=[]

Para X en lista:

compra.append(x.lower())

compra

esto es el resultado que querría:

compra=[["patata","garbanzos"],["melocotones","peras"],["leche","chocolate","galletas"]]

Ojalá podáis ayudarme, ya me he quedado sin ideas y soy algo nuevo en esto...

Modificar

Tenemos la lista con sublistas en mayúsculas.

```
listaMayusculas=[["PATATA","GARBABANZOS"],["MELOCOTONES","PERAS"],["LECHE","CHOCOLATE","
GALLETAS"]]


def recorrerLista(lista):
   listaNueva=[]
   for i in lista:
   if isinstance(i,list): # verificamos si es tipo lista
      listaNueva.append(recorrerLista(i)) # entra nuevamente a recorrerLista
   else:
      listaNueva.append(i.lower()) # ingresa elemento de lista en mayúscula
return listaNueva
```

Ahora la función recorrerLista(lista) nos devolverá una Lista nueva con todos sus elementos de la lista original modificados pero en el mismo orden y dentro de sus sublistas.

```
print("Lista nueva: ", recorrerLista(listaMayusculas))
```

Lista nueva: [['patata', 'garbabanzos'], ['melocotones', 'peras'], ['leche', 'chocolate', 'galletas']]
En cambio la Lista Original se mantiene igual sin modificación:

```
print("Lista original: ", listaMayusculas)
```

Lista original: [['PATATA', 'GARBABANZOS'], ['MELOCOTONES', 'PERAS'], ['LECHE', 'CHOCOLATE', 'GALLETAS']]