OBSERVACIONES DE LA PRACTICA

Juanita Gil - j.gila2@uniandes.edu.co - 202111556

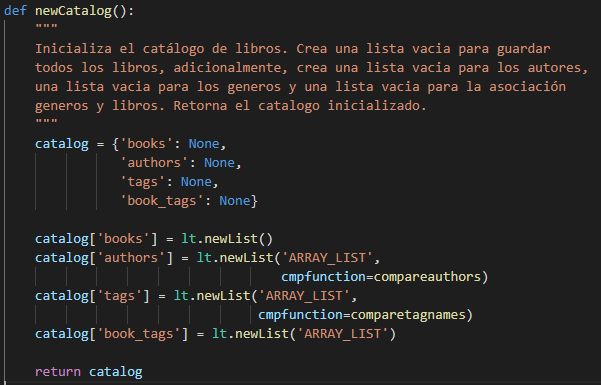
Gabriela Carvajal - g.carvajal@uniandes.edu.co - 202111058

1. ¿Cuáles son los mecanismos de interacción (I/O: Input/Output) que tiene el **view.py** con el usuario?

* El programa le pide al usuario (input) que ingrese por consola una opción correspondiente a su requerimiento (output) y depende de su elección se ejecutan funciones específicas que respondan a esta.

1. ¿Cómo se almacenan los datos de **GoodReads** en el **model.py**?

* La función “newCatalog()” inicializa el catálogo de los libros, luego crea una lista vacía para guardar estos dentro, también crea una lista vacía para los autores, otra para los géneros y otra para la asociación de libros. (Las listas de los autores y de los géneros son con arreglos y usan la función “cmpfunction” la cual compara los elementos de la lista)

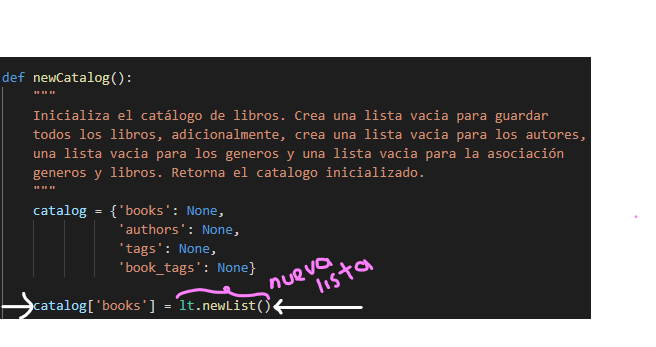


1. ¿Cuáles son las funciones que comunican el **view.py** y el **model.py**?

* “initCatalog”, “getBooksByAuthor”,“getBestBooks”, “countBooksByTag”, “loadData”. Las anteriores funciones se encuentran en el controller.py, el cual se encarga de comunicar el view.py y el model.py.

1. ¿Cómo se crea una lista?





* Las anteriores imágenes demuestran un ejemplo de cómo se crea una lista *(las flechas blancas lo señalan).* En este bloque de código primero se crea un diccionario “catalog” donde guarda “books, authors, tags y book\_tags”, luego en la última línea de código se está creando una nueva lista “books” con la función “newList” de DISCLib.ADT, es importante importar esto desde el incio del código para evitar errores al momento de ejecutarlo.

1. ¿Qué hace el parámetro **cmpfunction=None** en la función **newList()**?

* Esta función se encarga de comparar los elementos de la lista, además, el “None” provee una función de comparación de valor Key.

1. ¿Qué hace la función **addLast()**?

* Añade un elemento en la última posición de la lista, por lo tanto, también actualiza el apuntador a la última posición e incrementa el valor de la lista 1 unidad.

1. ¿Qué hace la función **getElement()**?

* Recorre la lista hasta encontrar el elemento que se desea buscar y retorna su posición, debe ser mayor que 0 y menor o igual al tamaño de la lista.

1. ¿Qué hace la función **subList()**?

* Retorna una sublista de la lista que se está examinando, esto se logra haciendo una copia de los datos que se desea extraer mediante la posición y el número de elementos.

1. ¿Observó algún cambio en el comportamiento del programa al cambiar la implementación del parámetro **“ARRAY\_LIST”** a **“SINGLE\_LINKED”**?

* Al cambiar a “single linked” la descarga de la información del catálogo es un poco más rápida en comparación a cuando estaba en “array list”, el resto de las opciones del menú no tuvieron cambios notables y seguían con una velocidad bastante buena.