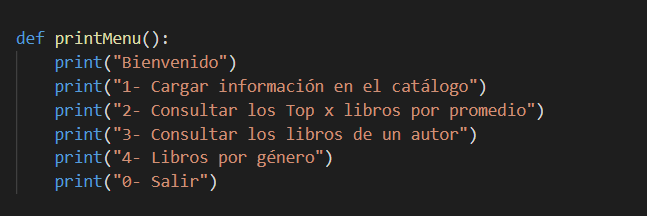
**OBSERVACIONES DEL LA PRACTICA**

Maicol Yojan Antonio Rincon Cod 202027329

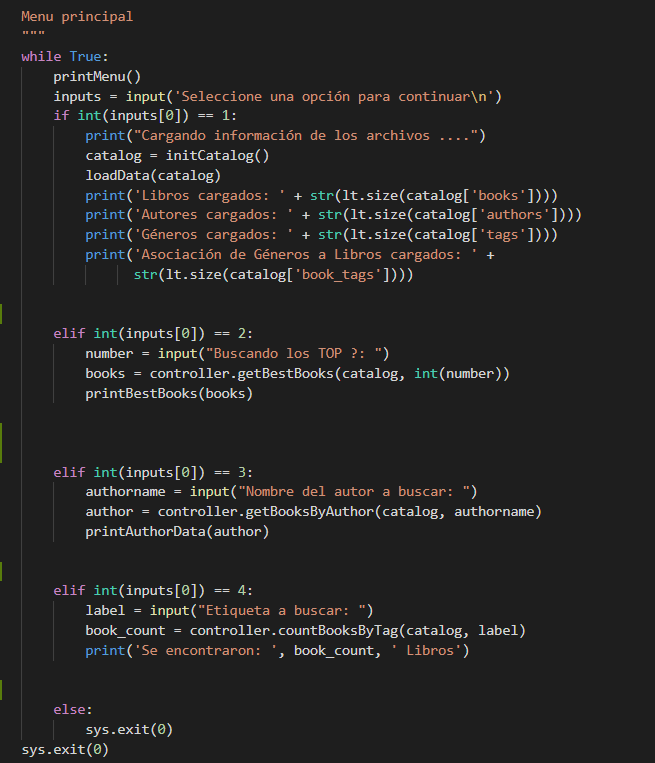
Lindsay Vanessa Pinto Morato Cod 202023138

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

Los mecanismos de interacción con el usuario propiamente dichos son los input y los print que se encuentran en el view. En este sentido, los mecanismos de salida que se encuentran son los que le muestra el sistema al usuario. En este caso se tiene:



Es de salida porque el usuario recibe esta información del sistema. Ahora bien, en el menú que se muestra a continuación, se encuentran mecanismos de interacción tanto de entrada (reciben opción digitada por el usuario, así como los datos que quiere consultar) y de salida (prints en el sistema con la información solicitada por el usuario).



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

Primero se crea una lista vacía para guardar todos los libros, adicionalmente, crea una lista vacía para los autores, una lista vacía para los géneros y una lista vacía para la asociación géneros y libros con el fin de retornar el catálogo inicializado. Esto se hace por medio de un ARRAY\_LIST. A esa lista vacía se le adicionan los diferentes datos de interés por medio de distintas funciones.

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

Todas las funciones del *view* se conectan al *model* por medio del *controller.* Por lo tanto, las funciones que comunican el *view* y el *model* serían todas las que se encuentran en el *controller*.

1. ¿Cómo se crea una lista?

Primero por medio de la función newList se crea una lista vacía la cual será complementada con las funciones addFirst y addLast. De igual forma se enriquece con las demás funciones asociadas que ayudan con la creación de elementos que servirán desde la identificación del nodo como los valores con los cuales se pueden encontrar las distintas posiciones.

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

El cmpfunction es la función de comparación que provee el usuario. En este caso como se está creando una lista vacía, no se tiene tampoco función de comparación, pero se crea este espacio desde el comienzo para ser utilizado posteriormente con la lista creada.

1. ¿Qué hace la funció **addLast()**?

Esta función agrega un elemento en la última posición de la lista.

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

Retorna el elemento en la posición de la lista. La posición se ingresa como parámetro en la función.

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

Esta función retorna una sublista de la lista inicial. Esta función retornará una lista con los elementos a partir de una cierta posición y con cierto número de elementos (ambos son recibidos como parámetro en la función) retornando una nueva lista después de hacer la copia de los elementos.

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

Al probar todas las opciones del programa tanto con **ARRAY\_LIST** como con **SINGLE\_LINKED** se observó un comportamiento similar. Al hacer pruebas con el tiempo de ejecución, para ambos tipos de TAD lista fueron iguales los tiempos de ejecución, difiriendo solo en cifras no significativas.