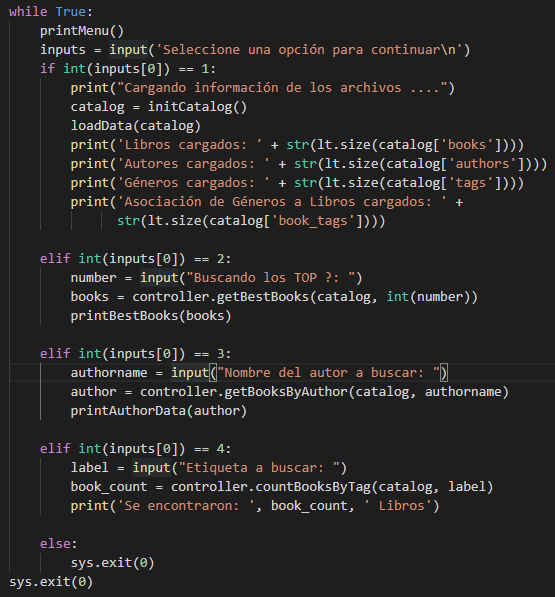
OBSERVACIONES DEL LA PRACTICA

Estudiante 1 María Alméciga Cod 202023369

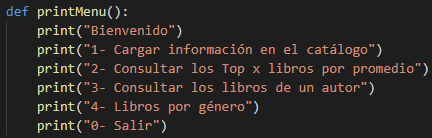
Estudiante 2 Andrés Vargas Cod 202013817

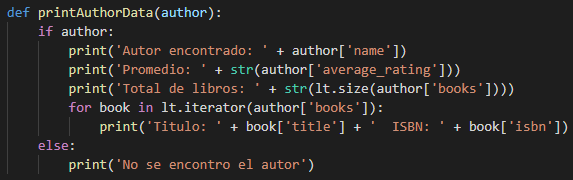
1. ¿Cuáles son los mecanismos de interacción (I/O: Input/Output) que tiene el **view.py** con el usuario?

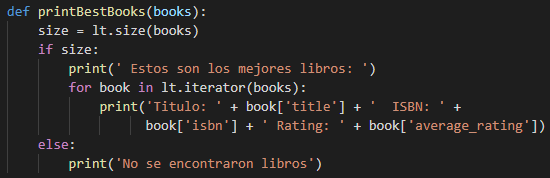
* Input: Los mecanismos de interacción son una serie de inputs en los que se solicita al usuario información sobre opciones para escoger, parámetros de búsqueda, y demás:



* Output: Los mecanismos de interacción son una serie de prints con la información de menús, información cargada, y demás:







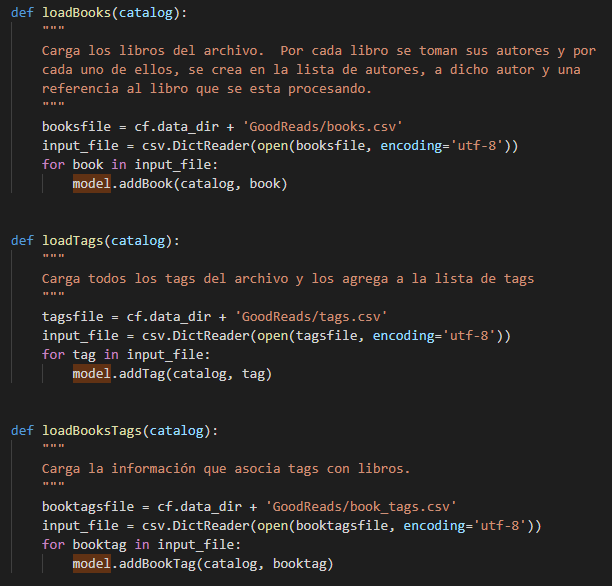
1. ¿Cómo se almacenan los datos de **GoodReads** en el **model.py**?

Algunas funciones en el model.py almacenan los datos, sacados del archivo CSV dado, en 4 listas para books, authors, tags, y book\_tags.

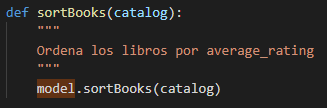
1. ¿Cuáles son las funciones que comunican el **view.py** y el **model.py**?

La mayoría de las funciones en el controller se encargan de comunicar o mediar entre el view y el model:

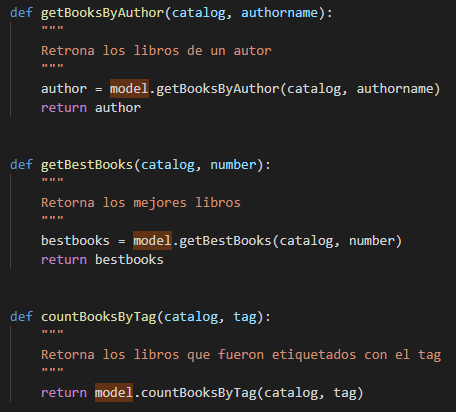
* Hay funciones del controller que interactúan con el model directamente para pedir información y posteriormente manipularla, y pasarla al view cuando sea requerido. Por ejemplo, las funciones de carga de datos:



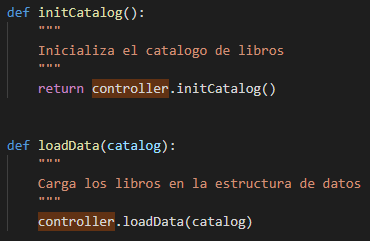
Las funciones de ordenamiento:

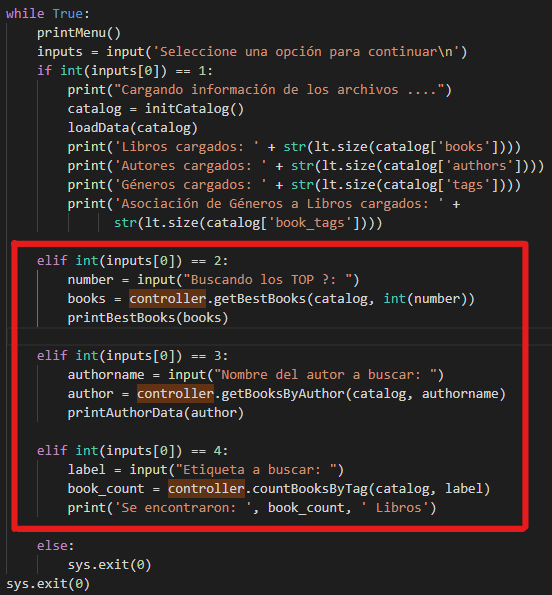


Las funciones de consulta sobre el catálogo:



* Estas funciones son solicitadas por el view cuando el usuario lo requiere, en funciones de este como:





1. ¿Cómo se crea una lista?

Las listas se crean mediante la función newList(). Esta crea por defecto una lista vacía, o si se pasa el nombre de un archivo en el parámetro filename, se crea una lista a partir de los elementos encontrados en el archivo en cuestión.

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

En el parámetro el usuario puede especificar una función para comparación de los elementos en la lista a crear. Si no se especifica, se utilizará la función predeterminada.

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

Recibiendo como parámetros una lista y un elemento, adiciona este elemento en la última posición de la lista, por lo que su tamaño incrementa en 1.

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

Recibiendo como parámetros una lista y un entero “pos” (valor entre 0 y el tamaño de la lista), recorre la lista hasta llegar al elemento en la posición pos, y lo retorna sin eliminarlo de la lista.

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

Recibiendo como parámetros una lista, un entero “pos” y un entero “numelem”, retorna una sublista de la lista original, empezando desde la posición pos de la misma, hasta completar la cantidad de elementos numelem.

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

No fue observado algún cambio evidente entre las dos implementaciones.