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

Tiene como input obligatorio la opción escogida en el menú, y dependiendo de esta, puede tener otros inputs como la cantidad de libros mostrada en el ranking, el autor y el género. Y de outputs tiene todo lo que se muestra dependiendo de la opción escogida.

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

Se crea un catálogo en donde están las tres listas donde cada una de estas representa los libros, los autores, y los géneros. Además, se genera una lista extra para la asociación de géneros y libros.

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

Básicamente el view.py solo importa el controller.py, por esto, es que no hay ninguna función que comunique directamente el view.py y el model.py. Ya que, el view.py se comunica con el controller.py, y este con el model.py.

**¿Cómo se crea una lista?**

Por medio de la función *newList*, la cual tiene 5 argumentos y retorna la propia lista. Estos argumentos son, datastructure que va a determinar si será un ARRAY\_LIST o un SINGLE\_LINKED, el cmpfunction; es para llamar una función que está definida y compara los elementos según ciertas categorías, el key; es el valor que se usa para usar la comparación por defecto ,el filename; para generar la listo con los elementos de algún archivo, el delimiter; si la lista proviene de un archivo argumento será el valor para separar los campos.

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

Este parámetro hace que posteriormente se pase una key para usarla comparación por defecto. Pero, en caso de que cmpfunction!=None, significa que el valor de key debe estar vacío porque cmpfunction ya debe dar los parámetros por lo cual se va a comparar.

**¿Qué hace la función *addLast()*?**

Esta función añade un elemento a la última posición de la lista.

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

Esta función recibe como parámetro una posición y una lista, y retorna el elemento que está en la posición y lista determinada.

**¿Qué hace la función *subList()*?**

Esta función retorna una sublista de la lista, recibiendo la posición y la cantidad de elementos de la futura sublista.

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

No se observó ningún cambio.