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

R/: En principio se le muestra el menú y se le pide ingresar la opción deseada, después del usuario escoger la opción deseada, se le solicitan los datos necesarios para ejecutar tal acción para seguido a eso ejecutarla y si mostrarle los resultados de la ejecución.

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

R/: Los datos se almacenan en un catálogo de listas de forma "SINGLE\_LINKED" , inicializando las listas vacías y agregando los datos de los archivos csv. a medida que se necesiten.

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

R/: Ninguna, puesto que la vista y el modelo no se comunican directamente.   
La funcionalidad depende del controlador que si tiene las funciones que conecta todas Las partes.

* ¿Cómo se crea una lista?

R/: importando el TAD Lista para de este usar la función newList() que además de crear la lista permite especificar las condiciones con las que se va crear .

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

R/: Especifica que la lista a crear no va tener parámetro alguno de comparación u ordenamiento.

* ¿Qué hace la función **addLast()**?

R/: Agrega un elemento a la última posición de una lista, tanto el elemento como la lista se pasan por parámetro.

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

R/: Retorna el elemento que se ubica en una posición especifica pasada por parámetro.

* ¿Qué hace la función **subList()**?

R/: Retorna una sección de la lista, es decir crea una nueva lista con solo los datos que se desean consultar.

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

R/: En el tiempo de ejecución a la hora de cargar los datos e interactuar con ellos , se sentían leves mejoras al usar “SINGLE\_LINKED” .