

- ¿Cuáles son los mecanismos de interacción (I/O: Input/Output) ¿Qué tiene el [view.py](#) con el usuario?
RTA: el view nos da la opción de crear el menú que le permite al usuario seleccionar la opción deseada y con base a esto retornar una función diferente como respuesta a la solicitud del usuario
- ¿Cómo se almacenan los datos de Goodreads en el [model.py](#)?
RTA: los datos se almacenan a través de funciones que agregan datos al catálogo, estas funciones contienen etiquetas llamadas tags que nos permiten organizar la información correspondiente que nos da Goodreads
- ¿Cuáles son las funciones que comunican el [view.py](#) y el [model.py](#)?
RTA: para esta relación se utilizan aquellas funciones que tienen que ver con la presentación y manipulación de datos, tales como cargar libros y etiquetas
- ¿Cuál es la función que permite crear una lista?, ¿Qué datos son necesarios?
RTA: NewCatalog(), la cual necesita los parametros: dataestructure, cmpfuncion
- ¿Para qué sirve el parámetro dataestructure en la función newList()?, ¿Cuáles son los posibles valores para este parámetro?
RTA: Especifica la estructura que se va a utilizar en la funcion
- ¿Para qué sirve el parámetro cmpfuncion en la función newList()?
RTA: Establece la función de comparación para los elementos de la lista.
- ¿Qué hace la función addLast()?
RTA: adiciona un elemento en la última posición de la lista y se actualiza el apuntador a la última posición. Se incrementa el tamaño de la lista en 1
- ¿Qué hace la función getElement()?
RTA: Recorre la lista hasta el elemento pos, el cual debe ser mayor que cero y menor o igual al tamaño de la lista. Se retorna el elemento en dicha posición sin eliminarlo.
- ¿Qué hace la función subList()?
RTA: Retorna una lista que contiene los elementos a partir de la posicion pos, con una longitud de numelem elementos. Se crea una copia de dichos elementos y se retorna una lista nueva.