OBSERVACIONES DE LA PRACTICA

Ana Sofía Castellanos 202114167

Martín Santiago Galván Castro 201911013

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

|  |  |  |
| --- | --- | --- |
| Posición | Input | Output |
| Menu principal | Pide al usuario el npumero de la opción que desea ejecutar | Muestra al usuario las opciones de las funciones que puede elegir.  Dirige al usuario a la opción elegida. |
| Cargar la información | No solicita nada al usuario | Muestra un mensaje con la siguiente información:  “””  Cargando información de los archivos ....  Libros cargados: \_\_\_  Autores cargados:\_\_\_  Géneros cargados: \_\_\_  Asociación de Géneros a Libros cargados: \_\_\_  “”” |
| Top x | Solicita la cantidad de libros que se desea consultar en la lista de Ranking | Muestra un mensaje con los títulos de la cantidad de libros solicitados por el usuario que se ubican en el top:  “””  Estos son los mejores libros:  Titulo: \_\_\_\_\_\_  Titulo: \_\_\_\_\_\_  …  “”” |

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

Por medio de la función newCatalog(), se crea un catalogo de libros en donde se almacena la información. Este catologo es un diccionario con las llaves: ‘books’, authors’´, ‘tags’, y ‘book\_tags’. Dentro de estas llaves se crean listas TAD. En el documento sin editar, todas las listas son declaradas como arreglos menos libros, la cual no tiene declarado un tipo de estructura de datos.

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

Las funciones que comunican view.py con model.py son las funciones que se encuentran en controller.py. Mas especificamente, las funciones de view.py llaman las funciones que se encuentran en controller.py, las cuales a su vez, llaman las funciones del modelo.

En view.py, las funciones initCatalog() y loadData(catalog) usan funciones que se encuentran en el controlador, controller.py, estas funciones del controlador usan a su vez funciones del modelo. Adicionalmente, en view.py, se usan funciones del controlador que no estan contenidas en otras funciones, estas son getBestBooks(…), getBooksByAuthor(…), countBooksByTag(…) y estas funciones comunican el controlador con el modelo.

1. ¿Cómo se crea una lista?

En el archivo list.py, se crean las listas como una lista vacia, especificamente como una lista simplemente encadenada por default. Esto se hace con la función newList(). Esta función hace uso de otra función de un archivo llamado liststructure.py en la carpeta DataStructures. Entre los parametros de la función, se encuentra ‘datastructure’ la cual se especifica como ‘SINGLE\_LINKED’, ‘cmpfunction’ = None, key = None, filename = None, delimiter=’,’.

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

Según la documentación, cmpfunction se refiere a la función de comparación para los elementos de la lista. Si no se provee función de comparación se utiliza la función por defecto. En esta caso, dado a que se especifica None para la función de comparación, no se realiza comparación para los elementos de la lista.

1. ¿Qué hace la funció **addLast()**?

Según la documentación, la función addLast(…) adiciona un elemento en la última posición de la lista y se actualiza el apuntador a la ultima posición. Se incrementa el tamaño de la lista en 1

1. ¿Qué hace la función **getElement()**?
2. ¿Qué hace la función **subList()**?
3. ¿Observó algún cambio en el comportamiento del programa al cambiar la implementación del parámetro **“ARRAY\_LIST”** a **“SINGLE\_LINKED”**?