EDA Lab 3

David Fuquen 202021113

Juan Andrés Eslava 202012035

PASO 3:

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

Hay múltiples outputs dentro del view. En particular, múltiples prints que le permiten al usuario saber qué información le está dando la máquina y qué puede hacer con la misma. Estos se encuentran en las siguientes funciones:

* printMenu en la que se le muestra las posibles opciones al usuario.
* printAuthorData en el que se muestra el autor encontrado, el promedio del puntaje de sus libros, el total de libros (en caso de no haber encontrado el autor, también se le avisa al usuario que el autor no fue encontrado).
* printBestBooks en la cual se muestran cuales fueron los mejores libros.
* Si se elige la opción 1 dentro del menú, se imprimen los libros cargados, los autores cargados, los géneros cargados y la asociación de géneros a libros cargados.

Además hay varios inputs en los que el usuario puede interactuar con el programa. En particular:

* Para elegir la opción que se desea ejecutar dentro del menú.
* Para elegir la cantidad de libros Top al seleccionar la segunda opción del menú. Esta información será usada para la función BestBooks.
* Para elegir el nombre del autor del cuál se requiere conseguir información si se selecciona la tercera opción del menú. Esta información se usará para la función BooksByAuthor.
* Para elegir la etiqueta a buscar si se seleccione la opción 3 del menú. Esta información se usará en la función BooksByTag

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

Los datos de los GoodReads se almacenan por medio de una función definida como catálogo, que es un diccionario cuyas llaves son ‘books’, ‘authors’, ‘tags’ y ’book\_tags’, y cuyos valores son listas de tipo ‘ARRAY\_LIST’ con la respectiva información acerca de libros, autores, tags y book tags.

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

import controller => En la vista, se importa el controlador, que es quién media entre la vista y el modelo, en este caso, para cualquier entrada que la vista obtenga del usuario, la vista llama al controlador, quien a su vez se encarga de llamar al modelo, por medio de import model, para poder ejecutar de manera correcta la opción marcada por el usuario. Por lo que en conclusión, la vista y el modelo se comunican a través del controlador.

PASO 4:

4-) ¿Cómo se crea una lista?

Para crear una lista se utiliza la función newList, en la cual se crea una nueva lista a partir de los parámetros datastructure (tipo de lista), cmpfunciton (función de comparación), key (identificador con el cual se comparan los elementos de la lista), filename (nombre del archivo, un csv de tipo utf8), delimeter (signo con el cual se espera se separen los elementos del csv). Con estos elementos, se utiliza la función propia del TAD lista newList para crear la lista con los datos previamente mencionados.

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

Cmpfunction es la función de comparación para los elementos de la lista. En este caso, al darle valor de None, se utiliza la función por defecto, pero se le debe dar un valor al parámetro Key.

6-) ¿Qué hace la función addLast()?

La función addLast añade un elemento a la última posición de una lista. Para esto, toma una lista y el elemento que se quiere añadir, y utiliza la función propia del TAD lista, addLast.

7-) ¿Qué hace la función getElement()?

La función get element() recibe como parámetros la dirección de la lista que se quiere evaluar, y la posición de algún elemento en la lista dada anteriormente, y la función retorna el elemento de la lista en la posición dada.

8-) ¿Qué hace la función subList()?

La función subList crea una sublista a partir de la lista original. Para esto, se toma la lista original, la posición en la lista sobre la cual se desea hacer inicio a la creación de la sublista, y la cantidad de elementos a copiar en la sublista. Con esta información, se usa la función propia del TAD lista subList.

PASO 5:

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

Al cambiar la forma en la que se almacenan los datos de los tags y book tags al tipo SINGLE\_LINKED, a la hora de correr el programa no se vió un gran cambio, sin embargo, su almacenamiento en memoria es diferente, por lo que puede que, para algunas operaciones como addFirst o addLast, puede ser más conveniente tener el tipo ARRAY\_LIST o SINGLE\_LINKED, pero en el caso de las opciones del menú del programa, al consultar los top 5 libros, o los libros del autor J.K. Rowling, el resultado fue el mismo.