OBSERVACIONES DEL LA PRACTICA

Estudiante 1 Cod 201924116

Estudiante 2 Cod 201922994

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

El usuario interactúa a través de un menú que funciona gracias a una función dentro de un ciclo que solo acaba cuando el usuario lo indica. Esta función reiterativa llama a las demás funciones que se definieron previamente. Estas funciones que son llamadas, a su vez, llaman funciones en el controlador. Luego de eso, cada función respectiva en el controldor llama otra función en el modelo, donde están los datos organizados como un tipo abstracto de datos.

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

Se almacenan como un tipo abstracto de datos (ADT), en particular una lista.

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

En view.py, initCatalog() llama al controlador y una función homonomina llama la función newCatalog() en el modelo.

En view.py, loadData(catalog) llama al controlador y una función llamada loadBooks(catalog) llama al modelo, en particular la función addBook(catalog, book).

Además, loadData(catalog) llama las funciones loadTags(catalog) y loadBookTags(catalog), las cuales llaman en el modelo a las funciones addTag(catalog, book) y addBookTag(catalog, book), respectivamente.

1. ¿Cómo se crea una lista?

Lista = lt.newList('ARRAY\_LIST', cmpfunction=”función que compara”)

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

Ese parametro indica la función que se va a usar para comparar. Es decir, es una función que sabe cómo deben estar organizados los datos, de manera que puede compararlos para después ordenarlos con otra función.

1. ¿Qué hace la funció **addLast()**?

Agrega un elemento a la lista en la última posición.

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

Esta función busca un elemento y lo recupera de la lista.

1. ¿Qué hace la función **subList()**?

Crea una “sublista” en la cual se ordenan los datos de otra forma, pero no se deben volver a guardar en memoria.

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

Con el parámetro SINGLE\_LINKED la función de cargar los datos se demoró mucho más que con ARRAY\_LIST. MUCHO.