OBSERVACIONES DEL LA PRACTICA

Estudiante 1 Jose Vicente Rincón Célis - Cod 202013321

Estudiante 2 Juan Esteban Arboleda - Cod 201921579

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

Los mecanismos de interacción con el usuario hacen parte de la función menú que se encuentra en el view.py

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

Para almacenar los datos primero tenemos que crear las listas vacías que se usaran para contener la información, listas que son creadas a partir de la siguiente función que crea los catálogos

Texto

Descripción generada automáticamente

Ahora si bien estas listas estan vacias necesitan contener la informacion ya clasificada por eso se usa la siguiente función:

Texto

Descripción generada automáticamente

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

Estas funciones para comunicar el view con el model se encuentran en el controller, dentro del archivo hay varios funciones, pero las que nos interesan son estas dado que están conectadas con el menu encontrado en el view y solicita la información en el model para que este la busque y la devuelva como un output en la consola

Texto

Descripción generada automáticamente

1. ¿Cómo se crea una lista?

La lista se crea a través de la función newList(…). Esta función no crea la lista como tal, si no que llama una función de liststructures.py (lt.newList(…)).



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

El parámetro define una función para comparar los elementos de la lista. El valor “default” del parámetro es None. Sin embargo, cuando se pasa None a cmpfunction se asigna una función predeterminada más adelante en la implementación.

Función predeterminada:



1. ¿Qué hace la funció **addLast()**?

La función addLast(…) añade un elemento en la última posición de la lista. Esta función no crea la lista como tal, si no que llama una función de liststructures.py (lt.addLast(…))



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

La función getElement(lst, pos) retorna el elemento en la posición pos de la lista. Esta función no crea la lista como tal, sino que llama una función de liststructures.py (lt.getElement(…)).



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

La función subList(lst, pos, numelem) crea una nueva lista (sub lista), con los elementos de lst en las posiciones entre pos y pos + numelem. Es decir, crea una sub-lista de lst, con los elementos desde pos, con una longitud numelem.

Nuevamente, esta función no crea la lista como tal, sino que llama una función de liststructures.py (lt.subList(…)).



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

El tiempo que tarda cargar los archivos aumentó algunos segundos al cambiar de ARRAY\_LIST a SINGLE\_LIST, pero el cambio no fue muy significativo.