Muhtacin Manik

12/2/19

CSC 2100

Short Description

For this project, we were asked to create a virtual library that holds books, magazines, and journals. To implement this I created a base class of Library and had three derived classes for books, magazines, and journals. I created a Catalog.txt file that held at least five books, magazines, and journals and then I created a function that reads in the data from the txt file to three vectors of books, magazines, and journals using the push_back function. Each item had the same data members of Id, Name, and status and each were set with a constructor from the Library class. However, books had an additional member of author name, magazines had an additional member of issue number, and journals had an additional member of volume. These were all implemented as private members in each class.

Furthermore, for this application we were asked that it allows a user to search for items. So, I created a search function that was used throughout the application. It was used when a user wanted to checkout an item and return an item because in order to complete these two actions they had to know the item id number. To find the item id number the search function was utilized. For the search function I added four options that included options to search catalog by name, search catalog by Books, Journals, or Magazines, search catalog for book by author name,

and display the whole catalog. This listed all information related to each category as well as the status of each item if it was checked in or checked out.

Also, for this application there is a save catalog option which saves all information that is inside each vector. It uses each get function in each class and outputs it into the txt file. It distinguishes each item by its type for example it separates the books, journals, and magazines.

There are a few exception handling that were added throughout this application. For example, when adding a book if the user enters a book that already exists they will get an error message. Another exception handling is with the do while statement for the menu. I added an exception handling method through try catch and if a user inputs a character instead of an integer they will get an error. Throughout the application there are a lot of bool variables that I utilized to indicate if an item exists and if it does not exist. This helped the user get the correct input for each option.