LAB211 Assignment

Type:	**Long Assignment**
Code:	**J1.L.P0002**
LOC:	**500**
Slot(s):	**N/A**

Title

Book Store Management – Read and Write File.

Background

A Book Store needs a program to manage Book and Publisher information. With basic requirements such as creating a Book (or Publisher), displaying Book (or Publisher) information, and updating information, ... Books and Publishers' information are stored in a text or binary file (Book.dat).

Program Specifications

Build a Book Store management program. With the following basic functions:

1. Publishers' management
   1.1. Create a Publisher
   1.2. Delete the Publisher
   1.3. Save the Publishers list to file
   1.4. Print the Publisher list from the file.
2. Books management
   2.1. Create a Book
   2.2. Search the Book
   2.3. Update a Book
   2.4. Delete the Book
   2.5. Save the Books list to file.
   2.6. Print the Books list from the file.

Others- Quit.

Features:

This system contains the following functions:

- **Publishers' management – 200 LOC**
  - **Function 1: Create a Publisher- 50 LOC**
    - The program requires to input a piece of Publisher information including Publisher's Id, Publisher's Name, and Phone Number fields.
    - After the inputted information has sent, the system will check the data validation with the following conditions:
      - Publisher's Id has pattern "Pxxxxx", with xxxxx is five digits, and is not allowed to duplicate
      - Publisher's Name is a string and has a length from 5 to 30 characters.
      - The phone is a number string that has a length from 10 to 12
    - The program allows to ask user to go back to the main menu.
  - **Function 2: Delete Publisher information – 50 LOC**
    - The user enters the Publisher's Id from the keyboard

- If it does not exist, the notification "Publisher's Id does not exist" message is shown. Otherwise, the Publisher was deleted.
- The system will show the result of the delete action with two success or fail status.
- The program allows to ask user to go back to the main menu.

- **Function 5: Save to file - 50 LOC**
  - The application allows to write a list of the Publisher's information in the file (Publisher.dat).
  - The program allows to ask user to go back to the main menu.
- **Function 6: Print all lists from file – 50 LOC**
  - Some situation, the system allows to load Publishers' information from the file to show into data grid (recommended: all information that gets from file, should be store into the collection)
  - The data grid should be displayed Publisher's information order by Publisher's Name ascending.

❖ **Books management – 300 LOC**

- **Function 1: Create a Book – 50 LOC**
  - The program requires to input to input a piece of Book information including Book's Id, Book's Name, Book's Price, Quantity, Publisher's Id, and Status fields.
  - After the inputted information has sent, the system will check the data validation with the following conditions:
    - Book's Id has pattern "Bxxxxx", with xxxxx is five digits, and is not allowed to duplicate
    - Book's Name is a string and has length from 5 to 30 characters.
    - Book's Price is a real number and greater than 0.
    - Book's Quantity is an integer number and greater than 0.
    - Status is a string and has values: Available or Not Available values.
    - If Publisher's Id did not exist, the program shows the "Publisher's Id is not found" message.
  - The program allows to ask user to go back to the main menu.
- **Function 2: Search the Book – 50 LOC**
  - The application allows the user to search both Book's Name (a part of Book's Name field) and Publisher's Id fields.
  - The system requires the user entering a search string (a part of Book's name field) or Publisher's Id values. After, the system will be returned a list of Book information.
  - If the list of Book is null, the notification "Have no any Book" message will be shown. Otherwise, the list of Book information will be shown order by the Book's Name field.
  - The program allows to ask user to go back to the main menu.

- **Function 3: Update Book – 50 LOC**
  - The user should enter the Book's ID from the keyboard.
  - If it does not exist, the notification "Book's Name does not exist" message will be shown. Otherwise, the Book will be shown to the user editing with changing information.
    - Notes: If the Information is not inputted, then the old information will be not updated
  - The system will show the result of the update action with success or failure status.
  - The program allows to ask user to go back to the main menu.
- **Function 4: Delete Book information – 50 LOC**
  - The user should enters the Book's ID from the keyboard
  - If it does not exist, the notification "Book's Name does not exist" message will be shown. Otherwise, the Book was deleted.
  - The system will show the result of the delete action with success or failure status.
  - The program allows to ask user to go back to the main menu.
- **Function 5: Save to file - 50 LOC**
  - The application allows to write a list of the Book's information in the file (Book.dat).
  - The program allows to ask user to go back to the main menu.
- **Function 6: Print all lists from file – 50 LOC**
  - Some situation, the system allows to load list Book information from the file to show into data grid (recommended: all information that gets from file, should be store into the collection)
  - The data grid should be displayed Book's information order by Book's Quantity descending.
    - If the Book has same Quantity then the system will be ordered by Book's Price ascending
    - Data grid should be formatted following

      Id Name Price Quantity Subtotal Publisher's Name Status
    - Note: SubTotal = BookPrice * Quantity
  - The program allows to ask user to go back to the main menu.
- **Function 7: Create a layout – 50 LOC**
  - The program is organized in the form of a function menu.
  - The support function will be asked if the user wants to continue or not.

- **Bonus 50 LOC if the student applies one of the Design Patterns (such as DAO pattern, Factory pattern, Repository pattern, and so on) in this project. More references for the design pattern:** https://www.tutorialspoint.com/design_pattern/index.htm
- The above specifications are only basic information; you must perform a requirements analysis step and build the application according to real requirements.
- The lecturer will explain the requirement only once in the first slot of the assignment.