

# ITEC2610 A/B/C Fall 2019

## Assignment 2

### To Submit:

1. AddBookDialog.java
2. Book.java
3. BookPropertyDialog.java
4. BookArrayModel.java
5. BookListWindow.java
6. BookStorage.java
7. UpdateBookDialog.java

### Instructions for File Submission:

Please put all 7 Java files into a .zip file **without using any folder** and then upload the .zip file through Moodle.

### Description:

A library in Toronto has contacted your company to develop a new book management system. As a new employee, you are asked by your team lead to create a simple demo program for the system. Your team lead has helped to construct a basic framework, and you need to complete the remaining work.

As shown in Figure 1, The UI designed for the demo includes a book list window and a property dialog window. With this interface, users can add, remove, update, and search for books.

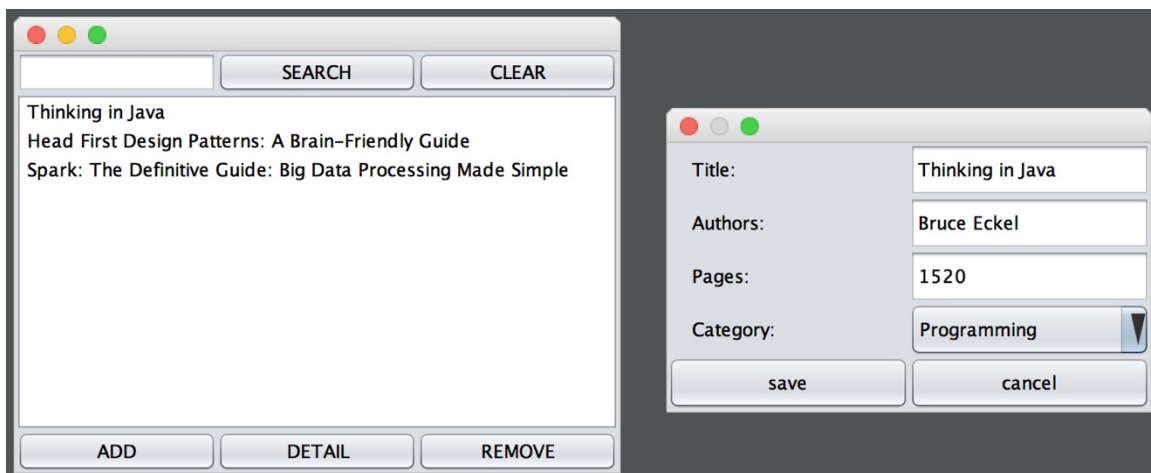


Figure 1. The book list window (left) and the property dialog window (right).

The required functionality of the demo is as follows.

1. When the application is started, the book list window will show up, and all the book titles will be listed in the ~~text-area~~ list in the middle of the window.
2. Users can enter a keyword in the text field on the top-left corner and then click the "SEARCH" button to search for books whose titles contain the input keyword. Now, only those matching book titles will show on the list ~~in the text-area~~.
3. The "CLEAR" button is designed to clean the search results. Specifically, when this button is clicked, the program will clear the keyword in the text field and list all the books again ~~in the text-area~~.
4. When the "ADD" button is clicked, the property dialog window will pop up. After filling up the fields (all the fields are required) in the dialog window, users can click the "SAVE" button to add the book. The "CANCEL" button can be used to abort the adding process and close the dialog window.
5. When an item is selected in the book list, users can click the "DETAIL" button to open a dialog window, which will show the detailed information of the selected book. This dialog window is almost the same as the one for adding, except that the "Title" field here is not editable. Users can edit the properties and click the "UPDATE" button to update the current book's properties.
6. Users can select an item in the book list and click the "REMOVE" button to remove the selected book.

Other instructions:

1. The main() method is in BookListWindow.java.
2. Please add your code only to where there's a "TODO" comment ((in AddBookDialog.java, BookListWindow.java, BookStorage.java, UpdateBookDialog.java and BookPropertyDialog.java) and leave other code unchanged.
3. The book titles are treated as unique identifiers and thus cannot be duplicated (i.e., no two books can have the same title) or changed once added.
4. You can assume that every book falls into one of the three categories: Programming, Database, and Design.
5. Please create more than 10 (and less than 100) books (with arbitrary titles/authors/pages/categories) in the initBooks() method of BookStorage.
6. Make sure that the JList for books only allows single selection.
7. If there is any problem with the input properties when adding/saving a book, pop up an error message dialog and then do nothing.
8. The look of the interface may vary slightly from one platform to another. The basic layout should be the same.