

Library Management

Rishi Dua, 2010EE50557

28 January, 2014

1 LIBRARY MANAGEMENT

1.1 PROBLEM STATEMENT

Write a program in Java only for Library Management.

1. Use a text file to store the following information about the books available in the library. (Title of the book) (Author name) (Publication) (Edition) Read from the file and store them as array of objects. Provide user the functionality to search for some books based on Book title, author name, publication or any combination of these.
2. Add the Library Incharge Login functionality to above program which enables him to add or remove any book from the library. The changes must be reflected in the Library record and all student accounts as well.
3. Add the Student Login functionality to above program to keep a record of the books issued to each student along with the date of issue.
4. Add the functionality to above program to calculate total fine imposed in case any user fails to deposit the issued book within a period of 1 week.

All objective optional functions are implemented.

1.2 ABSTRACT

In the real world, we often find many individual objects all of the same kind. There may be thousands of other bicycles in existence, all of the same make and model. Each book was

built from the same set of blueprints and therefore contains the same components. In object-oriented terms, we say that your book is an instance of the class of objects known as books. A class is the blueprint from which individual objects are create

1.3 SPECIFICATION AND ASSUMPTIONS

Tool Specifications:

Language used: Java

Platform: Ubuntu 12.04

Additional tools used: none

Eclipse Version: Version: 3.7.2

Assumptions

The input file already exists in the specified TSV format

Problem specifications

This is the complete list of members for LibManagement, including all inherited members.

addbook() LibManagement [static]

booklist (defined in LibManagement) LibManagement [package, static]

displaybooks() LibManagement [static]

i LibManagement [package, static]

loggedin (defined in LibManagement) LibManagement [package, static]

main(String[] args) (defined in LibManagement) LibManagement [static]

makelist() LibManagement [static]

removebook() LibManagement [static]

searchbooks() LibManagement [static]

Class book

Public Member Functions

Book (String[] parts)

String gettitle ()

String getauthor ()

String getpublication ()

String getedition ()

Package Attributes

String title

String author

String publication

String edition

Detailed Description

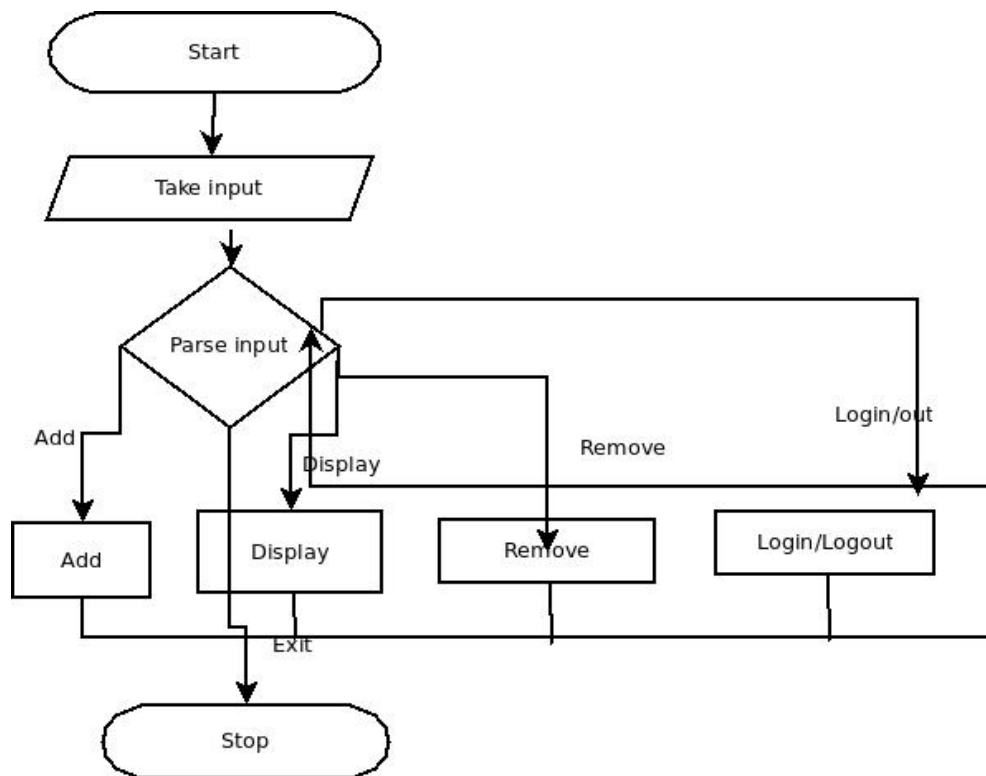
Constructor & Destructor Documentation

Book.Book (String[] parts)

Parameters:

parts

1.4 FLOW CHART



1.5 LOGIC IMPLEMENTATION

The problem is broken into the following 7 parts: 1: login

2: view all

3: search

4: add (admin only)

5: remove (admin only)

6: logout

7::exit

1.6 EXECUTION DIRECTIVE

Compiling: javac LibManagement.java

Running:

java LibManagement

Follow the on-screen instructions after that

1.7 OUTPUT OF THE PROGRAM

Library Manager

Using as Guest

Enter choice: 1: login, 2: view all, 3: search 4: add (admin only), 5: remove (admin only)

2

The list of books is as follows:

A D B C

E H F G

I L J K

Tit Ed Aut Pub

qwe rtytit1 wer ert

au2 random pub2 ed2

au dua pi edrishi

www www www www

random random random random

newnew newnew newnew newnew

rishi ed dua pub

Using as Guest

Enter choice: 1: login, 2: view all, 3: search 4: add (admin only), 5: remove (admin only)

3

Enter string to search

D

The book is A, ed: D by B. Publication: C

Using as Guest

Enter choice: 1: login, 2: view all, 3: search 4: add (admin only), 5: remove (admin only)

2

The list of books is as follows:

A D B C

E H F G

I L J K

Tit Ed Aut Pub

qwe rtytit1 wer ert

au2 random pub2 ed2

au dua pi edrishi

www www www www

random random random random

newnew newnew newnew newnew

rishi ed dua pub

Using as Guest

Enter choice: 1: login, 2: view all, 3: search 4: add (admin only), 5: remove (admin only)

1

```

Enter username
admin
Enter username
pass
Hello admin!
Logged in as admin
Enter choice: 1: login, 2: view all, 3: search 4: add (admin only), 5: remove (admin only)
5
The list of books is as follows:

ID 0 A, ed: D by B. Publication: C
ID 1 E, ed: H by F. Publication: G
ID 2 I, ed: L by J. Publication: K
ID 3 Tit, ed: Ed by Aut. Publication: Pub
ID 4 qwe, ed: rtytit1 by wer. Publication: ert
ID 5 au2, ed: random by pub2. Publication: ed2
ID 6 au, ed: dua by pi. Publication: edrishi
ID 7 www, ed: www by www. Publication: www
ID 8 random, ed: random by random. Publication: random
ID 9 newnew, ed: newnew by newnew. Publication: newnew
ID 10 rishi, ed: ed by dua. Publication: pub
Enter ID of book to delete
7
Logged in as admin
Enter choice: 1: login, 2: view all, 3: search 4: add (admin only), 5: remove (admin only)
7

```

1.8 RESULT

A Libray manager is developed
The code is developed with Java

The user is able to add/remove/view all books.
the user can login as admin. This gives him write privilidges to the database.

The documentation made using doxygen is also attached with the submission

1.9 CONCLUSION

Successfully developed a library manager. Key things learnt
Authentication is the act of confirming the truth of an attribute of a datum or entity. This might involve confirming the identity of a person or software program, tracing the origins of

an artifact, or ensuring that a product is what its packaging and labeling claims to be. Authentication often involves verifying the validity of at least one form of identification. This is used for login.

Reading and writing files from java using buffered readers is learnt. The details of reading, writing, creating, and opening files. There are a wide array of file I/O methods to choose from which were explored in this project.