

COM102 (Programming II - Java)

Coursework - Practical Skills Assessment 2

Title: Library Loans

Due: Friday 8th April 2022 at 3:00pm

This coursework constitutes 60% of the total marks for the module. Feedback will be made available on Blackboard within 20 working days from the submission deadline.

For this assignment you are normally expected to work in pairs.

As a pair, when submitting your assignment you are agreeing to the following statement:

I declare that this is all my own work and does not contain unreferenced material copied from any other source. I have read the University's policy on plagiarism and understand the definition of plagiarism. If it is shown that material has been plagiarised, or I have otherwise attempted to obtain an unfair advantage for myself or others, I understand that I may face sanctions in accordance with the policies and procedures of the University. A mark of zero may be awarded and the reason for that mark will be recorded on my file.

The University policy on **plagiarism** is available at

<https://www.ulster.ac.uk/student/exams-and-appeals/policies-procedures>

Module Learning Outcomes:

1. demonstrate a knowledge of the programming constructs and data types of the selected language
2. analyze a problem using an object-oriented approach
3. design and develop code in a professional manner that facilitates readability and maintainability
4. solve problems involving a logical component.

Assessment Criteria:

- appropriate use of variables, structured and readable code (20%)
- program executes according to specification (30%)
- implementation of suitable program control structures to solve problem (30%)
- structured approach to testing (20%)

Overview

This assignment is concerned with manipulating lists / arrays of objects. You will write a program to store and manage details of library items currently on loan to library users according to the following functional requirements:

- **Functional Requirements**

1. The set of lendable items held by the library includes two types, i.e. Book and Multimedia, and is predefined (read only) as given in file ITEMS.csv.

Note that an item's unique identifier is its library barcode since there may be more than one copy of an item in the library.

2. The set of library users is predefined (read only) as given in file USERS.csv.
3. The list of current (active) loans is maintained in the file LOANS.csv.
4. When the program is started, it should read from these three csv files to populate the application with any previously stored data as a starting point.

Within the program items, users and loans should be stored as lists / arrays of objects, with properties according to the column headings in the above files, i.e. there will be a list / array of items, a list / array of users and a list / array of loans.

5. The program should allow the librarian to **issue** an item (i.e. to create a loan). To do this, the librarian supplies the item barcode and the user id.

The system should check that the barcode exists and that the user id exists. For books, loans are for a four week period. For multimedia items, loans are for a one week period.

When an item is issued a loan object is created and added to the list / array of loans.

6. The program should allow the librarian to **renew** an existing loan.

On supply of the item barcode, the loan's return date is increased by two weeks from the current date for books and one week for multimedia items and the number of renewals is increased by one. A book cannot be renewed more than three times and the maximum number of renewals for a multimedia item is two.

7. The program should allow the librarian to record the **return** of an item on loan. To do this, the librarian supplies the barcode. Items are always returned on or before their due date.

When an item on loan is returned the corresponding loan object is removed from the list of loans.

8. The program should allow the librarian to **view** (a) all the items currently on loan and (b) all the items held by the library.

9. When the program exits, the list of current loans should be **written** to LOANS.csv.

10. You must develop a set of test cases and record the results of applying these tests to your software.

The program must employ a console interface only.

- **Other Assignment Requirements:**

The adoption of object-oriented principles should be evident in your implementation of the above requirements.

A structured approach to testing should be evidenced by submission of a test plan and outcomes in accordance with the above requirements.

A Vodcast encoded in mp4 format, lasting no more than **5 minutes**, is required to be produced to accompany your implementation within which a brief walk-through of your code is provided – it should demonstrate clearly the features you have created as well as a clear readable review of the code that make the features work. It is expected that all the team members should take part in the demonstration vodcast to report their contributions to the solution.

Submission

1. The assignment which is normally expected to be completed in pairs needs to be submitted electronically to BlackBoard on or before **Friday 8th April 2022 at 3:00pm** against the 'Practical Skills Assessment #2' assignment in the COM102 area.
2. The following documents need to be submitted
 - A zip file called **com102PSA2_YourGroupID.zip** containing the NetBeans project folder and all associated sub-folders for all source code (.java), compiled files (.class) and IDE project files.
 - One word document file containing the test plan and results
 - One pdf file containing all source code – this is for use by moderators
 - A **team declaration form** (available with this assignment specification on Blackboard) must be completed to indicate each team member's agreed percentage contribution to the submitted solution. **Your submission will not be awarded a mark unless a signed team declaration form is received.**
 - A copy of the vodcast should be submitted using the link **PSA2 Vodcast Submission** within the module area of Blackboard by the submission deadline (3:00pm on 8th April 2022). Please ensure the name of the submitted vodcast is given as **com102PSA2_yourGroupID_VODCAST.mp4**.
3. If the work is completed in pairs, then only **ONE** copy of the work should be submitted, with **both names and student IDs clearly marked on all the submitted work**.
4. Please ensure that you keep a secure electronic backup copy of your assignment.

***** IMPORTANT *** Marks Weighting for Pair Contributions:**

This assignment should normally be completed in pairs. If completed in pairs, then **each** of the pair must sign the 'team declaration' form to indicate what percentage contribution each member made to the overall effort. Where both in the team contribute equally in overall effort to the team solution, then there is **no** penalty for working in a team of two as opposed to working individually. Each person will obtain the full mark awarded to the submitted work.

For any pair submission deviating from an equal division of labour, marks for this assignment will be scaled down according to the declarations made. The scaling will happen according to the following example of the scheme: Consider a pair-submitted piece of work that is deemed worthy of a mark of 50%. Consider also that one of the pair has declared their contribution to be 60% and the other declared at 40% (i.e. 10/50 or 20% less effort than an equitable contribution). The person whose contribution is the lesser will have their mark reduced by 10/50 or 20%, i.e. to 40% in this example. The person whose contribution was set at 60% will receive the full awarded mark of 50%.

In the case of pair-submissions, only ONE copy of the solution is required with submitted source code files and test plans clearly indicating the names and student IDs of both team members.