# **Programming Assignment Unit 1**

Computer Science, University of the People

CS 1101-01 Programming Fundamentals - CS 1102-01 - AY2024-T2

Instructor, Noman Shihadeh

November 26, 2023

## Library system

For this assignment, we were asked to write a program that allows the user to manage a library system. The system will allow the user to do the following basic actions:

1. Add books.
2. Borrow books.
3. Return books.
4. List books (not required but seemed like it should be included)
5. Exit the program.

Source Code:

import java.util.Scanner;

public class LibrarySystem {

    // Constants and global variables

    private static final int MAX\_BOOKS = 100;

    private static String[] titles = new String[MAX\_BOOKS];

    private static String[] authors = new String[MAX\_BOOKS];

    private static int[] quantities = new int[MAX\_BOOKS];

    private static int bookCount = 0;

    private static Scanner scanner = new Scanner(System.in);

    public static void main(String[] args) {

        // Main loop to handle user choices

        while (true) {

            // Display the options menu

            System.out.println("\nOptions:\n" +

                    "1. Add Books\n" +

                    "2. Borrow Books\n" +

                    "3. Return Books\n" +

                    "4. List Books\n" +

                    "5. Exit");

            System.out.print("Choose an option: ");

            String choice = scanner.nextLine();

            // Handling user's choice

            switch (choice) {

                case "1":

                    addBooks();

                    break;

                case "2":

                    borrowBooks();

                    break;

                case "3":

                    returnBooks();

                    break;

                case "4":

                    listBooks();

                    break;

                case "5":

                    // Exit the application

                    System.out.println("Exiting...");

                    return;

                default:

                    // Invalid option handling

                    System.out.println("Invalid option. Please try again.");

            }

        }

    }

    // Method to add books to the library

    private static void addBooks() {

        // Check if the library is full

        if (bookCount >= MAX\_BOOKS) {

            System.out.println("The Library is full. Cannot add more books.");

            return;

        }

        // Input book details

        System.out.print("Enter book title: ");

        String title = scanner.nextLine();

        System.out.print("Enter author: ");

        String author = scanner.nextLine();

        System.out.print("Enter quantity: ");

        int quantity = Integer.parseInt(scanner.nextLine());

        // Check if the book already exists and update quantity

        int index = findBookIndex(title);

        if (index != -1) {

            quantities[index] += quantity;

            System.out.println("Quantity updated.");

        } else {

            // Add new book to the library

            titles[bookCount] = title;

            authors[bookCount] = author;

            quantities[bookCount] = quantity;

            bookCount++;

            System.out.println("Book added.");

        }

    }

    // Method to borrow books from the library

    private static void borrowBooks() {

        // Input book title and find it in the library

        System.out.print("Enter book title: ");

        String title = scanner.nextLine();

        int index = findBookIndex(title);

        // If book is not found, return

        if (index == -1) {

            System.out.println("Book not found.");

            return;

        }

        // Input quantity to borrow

        System.out.print("Enter quantity to borrow: ");

        int quantity = Integer.parseInt(scanner.nextLine());

        // Check if enough copies are available

        if (quantities[index] >= quantity) {

            quantities[index] -= quantity;

            System.out.println("Book borrowed.");

        } else {

            System.out.println("Not enough copies available.");

        }

    }

    // Method to return books to the library

    private static void returnBooks() {

        // Input book title and find it in the library

        System.out.print("Enter book title: ");

        String title = scanner.nextLine();

        int index = findBookIndex(title);

        // If book is not found, return

        if (index == -1) {

            System.out.println("Book not found.");

            return;

        }

        // Input quantity to return

        System.out.print("Enter quantity to return: ");

        int quantity = Integer.parseInt(scanner.nextLine());

        // Update library quantities

        quantities[index] += quantity;

        System.out.println("Book returned.");

    }

    // Method to list all books in the library

    private static void listBooks() {

        // Check if the library is empty

        if (bookCount == 0) {

            System.out.println("No books in the library.");

            return;

        }

        // Display all books

        System.out.println("Books in the library:");

        for (int i = 0; i < bookCount; i++) {

            System.out.println("Title: " + titles[i] + ", Author: " + authors[i] + ", Quantity: " + quantities[i]);

        }

    }

    // Helper method to find the index of a book by title

    private static int findBookIndex(String title) {

        for (int i = 0; i < bookCount; i++) {

            if (titles[i].equals(title)) {

                return i;

            }

        }

        return -1;

    }

}

Output:

Options:

1. Add Books

2. Borrow Books

3. Return Books

4. List Books

5. Exit

Choose an option: 1

Enter book title: Book 1

Enter author: Simon

Enter quantity: 5

Book added.

Options:

1. Add Books

2. Borrow Books

3. Return Books

4. List Books

5. Exit

Choose an option: 1

Enter book title: Book 2

Enter author: Bob

Enter quantity: 10

Book added.

Options:

1. Add Books

2. Borrow Books

3. Return Books

4. List Books

5. Exit

Choose an option: 1

Enter book title: Book 1

Enter author: Simon

Enter quantity: 2

Quantity updated.

Options:

1. Add Books

2. Borrow Books

3. Return Books

4. List Books

5. Exit

Choose an option: 4

Books in the library:

Title: Book 1, Author: Simon, Quantity: 7

Title: Book 2, Author: Bob, Quantity: 10

Options:

1. Add Books

2. Borrow Books

3. Return Books

4. List Books

5. Exit

Choose an option: 2

Enter book title: Book 1

Enter quantity to borrow: 2

Book borrowed.

Options:

1. Add Books

2. Borrow Books

3. Return Books

4. List Books

5. Exit

Choose an option: 3

Enter book title: Book 1

Enter quantity to return: 2

Book returned.

Options:

1. Add Books

2. Borrow Books

3. Return Books

4. List Books

5. Exit

Choose an option: 4

Books in the library:

Title: Book 1, Author: Simon, Quantity: 7

Title: Book 2, Author: Bob, Quantity: 10

Options:

1. Add Books

2. Borrow Books

3. Return Books

4. List Books

5. Exit

Choose an option: 5

Exiting...

The main parts of the code:

1. The “main” method is our program entry point, it is also where we show the user the options he has and process his input by calling the appropriate method relevant to his selection.
2. The “addBooks” method allows the user to add books to the library by guiding him through the adding process of entering the required values for a book. The data input by the user is validated and then added to our array storage for later use.
3. The “borrowBooks” method allows the user to borrow several books from the library by entering the title and amount of the book required. If the book isn’t in the library, he is informed that it is out of stock.
4. The “returnBooks” method allows the user to return the books to the library after borrowing them and adding the amount back to the array storage.
5. The “listBooks” method lists all the books in the library and their amounts. Note that this method was not required for the assignment but was useful to help monitor and debug the code.

## References

Java Language and Virtual Machine Specifications

<https://docs.oracle.com/javase/specs/index.html>

Introduction to Programming Using Java - Version 9.0, JavaFX Edition

<https://math.hws.edu/javanotes/>

Source Code:

