ASSIGMENT SUBMISSION

**Student Details**

**INFORMATION**

**Name : B.SAI YADHESHWARAN**

**Registration NO : 2117240020328**

**Department : CSE**

**Section : F**

**Subject Code : CS23312**

**Subject Name : Object Oriented Programming**

**Assignment No. : 1**

1. **AIM:**

To create a method in the Member class that displays only the books currently borrowed by the member in a library management system.

1. **ALGORITHM:**

* **Start.**
* **Define a Book class with attributes:**
* **title (String)**
* **isBorrowed (boolean)  
  and methods: borrow(), returnBook(), isBorrowed(), and getTitle().**
* **Define a Member class with attributes:**
* **name (String)**
* **borrowedBooks (List of Book objects).**

** In the Member class:**

* **Create a method borrowBook(Book) to add a book if it is not already borrowed.**
* **Create a method returnBook(Book) to return a borrowed book.**
* **Create a method displayBorrowedBooks() that:**
  + **Prints the member’s name.**
  + **If the list of borrowed books is empty, print “No books currently borrowed.”**
  + **Otherwise, loop through the list and print each borrowed book’s title.**

** In the main() method:**

* **Create some Book objects.**
* **Create a Member object.**
* **Borrow and return books using the methods.**
* **Call displayBorrowedBooks() to check the output.**

** Stop.**

**JAVA CODE:**import java.util.ArrayList;

import java.util.List;

class Book {

    private String title;

    private boolean isBorrowed;

    public Book(String title) {

        this.title = title;

        this.isBorrowed = false;

    }

    public String getTitle() {

        return title;

    }

    public boolean isBorrowed() {

        return isBorrowed;

    }

    public void borrow() {

        isBorrowed = true;

    }

    public void returnBook() {

        isBorrowed = false;

    }

}

class Member {

    private String name;

    private List<Book> borrowedBooks;

    public Member(String name) {

        this.name = name;

        this.borrowedBooks = new ArrayList<>();

    }

    public void borrowBook(Book book) {

        if (!book.isBorrowed()) {

            book.borrow();

            borrowedBooks.add(book);

            System.out.println(name + " borrowed: " + book.getTitle());

        } else {

            System.out.println(book.getTitle() + " is already borrowed.");

        }

    }

    public void returnBook(Book book) {

        if (borrowedBooks.contains(book)) {

            book.returnBook();

            borrowedBooks.remove(book);

            System.out.println(name + " returned: " + book.getTitle());

        }

    }

    public void displayBorrowedBooks() {

        System.out.println("Currently borrowed books by " + name + ":");

        if (borrowedBooks.isEmpty()) {

            System.out.println("No books currently borrowed.");

        } else {

            for (Book book : borrowedBooks) {

                System.out.println("- " + book.getTitle());

            }

        }

    }

}

public class LibraryDemo {

    public static void main(String[] args) {

        Book b1 = new Book("Java Programming");

        Book b2 = new Book("Python Basics");

        Book b3 = new Book("Data Structures");

        Member m1 = new Member("Alice");

        m1.borrowBook(b1);

        m1.borrowBook(b2);

        m1.displayBorrowedBooks();

        m1.returnBook(b1);

        m1.displayBorrowedBooks();

}

}

**OUTPUT:**

Alice borrowed: Java Programming

Alice borrowed: Python Basics

Currently borrowed books by Alice:

- Java Programming

- Java Programming

- Python Basics

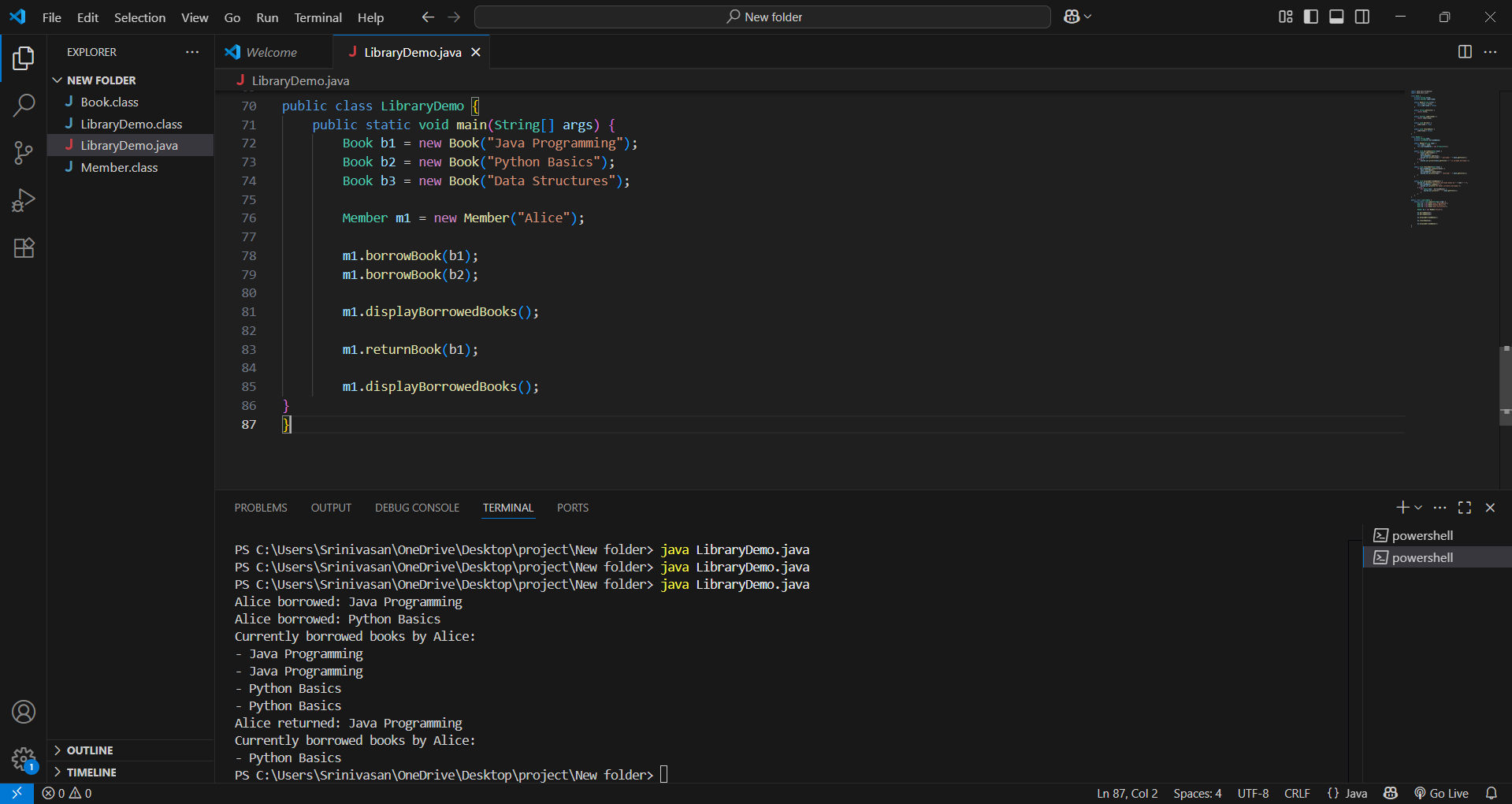
- Python Basics

Alice returned: Java Programming

Currently borrowed books by Alice:

- Python Basics

Screenshots:



**GitHub Repository Link:**