Task 3

1.Answer

package com.task3;

class Book {

private int bookID;

private String title;

private String author;

private boolean isAvailable;

public Book(int bookID, String title, String author) {

this.bookID = bookID;

this.title = title;

this.author = author;

this.isAvailable = true; // By default, book is available

}

public int getBookID() {

return bookID;

}

public String getTitle() {

return title;

}

public String getAuthor() {

return author;

}

public boolean isAvailable() {

return isAvailable;

}

public void setAvailable(boolean available) {

isAvailable = available;

}

*@Override*

public String toString() {

return "Book ID: " + bookID + ", Title: " + title + ", Author: " + author + ", Available: " + isAvailable;

}

}

class Library {

private Book[] books;

private int capacity;

private int count;

public Library(int capacity) {

this.capacity = capacity;

this.books = new Book[capacity];

this.count = 0;

}

public void addBook(Book book) {

if (count < capacity) {

books[count++] = book;

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

} else {

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

}

}

public void removeBook(int bookID) {

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

if (books[i].getBookID() == bookID) {

books[i].setAvailable(false);

System.***out***.println("Book removed successfully.");

return;

}

}

System.***out***.println("Book with ID " + bookID + " not found.");

}

public Book searchBook(int bookID) {

for (Book book : books) {

if (book != null && book.getBookID() == bookID) {

return book;

}

}

return null;

}

public void displayBooks() {

if (count == 0) {

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

return;

}

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

for (Book book : books) {

if (book != null && book.isAvailable()) {

System.***out***.println(book);

}

}

}

}

Main class

package com.task3;

import java.util.Scanner;

public class Mainclass {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.***in***);

Library library = new Library(10); // Creating library with capacity for 10 books

while (true) {

System.***out***.println("\nLibrary Management System Menu:");

System.***out***.println("1. Add Book");

System.***out***.println("2. Remove Book");

System.***out***.println("3. Search Book");

System.***out***.println("4. Display All Books");

System.***out***.println("5. Exit");

System.***out***.print("Enter your choice: ");

int choice = scanner.nextInt();

scanner.nextLine(); // Consume newline character

switch (choice) {

case 1:

System.***out***.print("Enter Book ID: ");

int bookID = scanner.nextInt();

scanner.nextLine(); // Consume newline character

System.***out***.print("Enter Title: ");

String title = scanner.nextLine();

System.***out***.print("Enter Author: ");

String author = scanner.nextLine();

library.addBook(new Book(bookID, title, author));

break;

case 2:

System.***out***.print("Enter Book ID to remove: ");

int removeID = scanner.nextInt();

library.removeBook(removeID);

break;

case 3:

System.***out***.print("Enter Book ID to search: ");

int searchID = scanner.nextInt();

Book foundBook = library.searchBook(searchID);

if (foundBook != null) {

System.***out***.println("Book found: " + foundBook);

} else {

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

}

break;

case 4:

library.displayBooks();

break;

case 5:

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

System.*exit*(0);

default:

System.***out***.println("Invalid choice. Please enter a number between 1 and 5.");

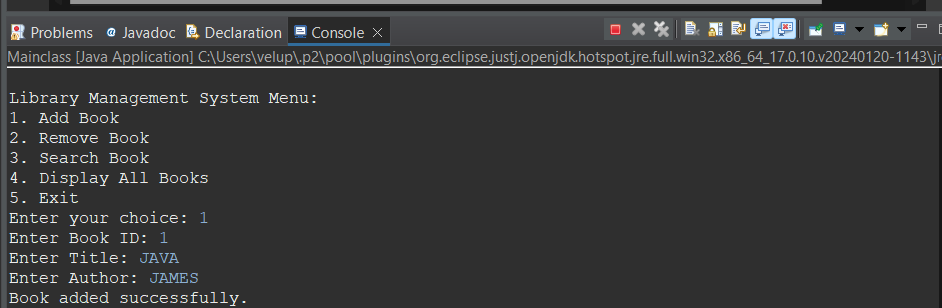
}

}

}

}

Output



2.

package com.task3;

public interface Taxable {

double ***salesTax*** = 0.07; // 7%

double ***incomeTax*** = 0.105; // 10.5%

double calcTax(); // Abstract method

}

// Class Employee implementing Taxable

class Employee implements Taxable {

private int empId;

String name;

private double salary;

// Constructor

public Employee(int empId, String name, double salary) {

this.empId = empId;

this.name = name;

this.salary = salary;

}

// Implementing abstract method to calculate income tax on yearly salary

*@Override*

public double calcTax() {

return salary \* ***incomeTax***;

}

}

// Class Product implementing Taxable

class Product implements Taxable {

private int pid;

private double price;

private int quantity;

// Constructor

public Product(int pid, double price, int quantity) {

this.pid = pid;

this.price = price;

this.quantity = quantity;

}

// Implementing abstract method to calculate sales tax on unit price of product

*@Override*

public double calcTax() {

return price \* quantity \* ***salesTax***;

}

}

Main class

package com.task3;

public class DriverMain {

public static void main(String[] args) {

// Creating employee object

Employee employee = new Employee(1, "Ramya", 50000.0);

// Creating product object

Product product = new Product(101, 10.0, 5);

// Printing income tax and sales tax respectively

System.***out***.println("Income Tax for Employee " + employee.name + ": $" + employee.calcTax());

System.***out***.println("Sales Tax for Product: $" + product.calcTax());

}

}

Output:

