JAVA DAY-10

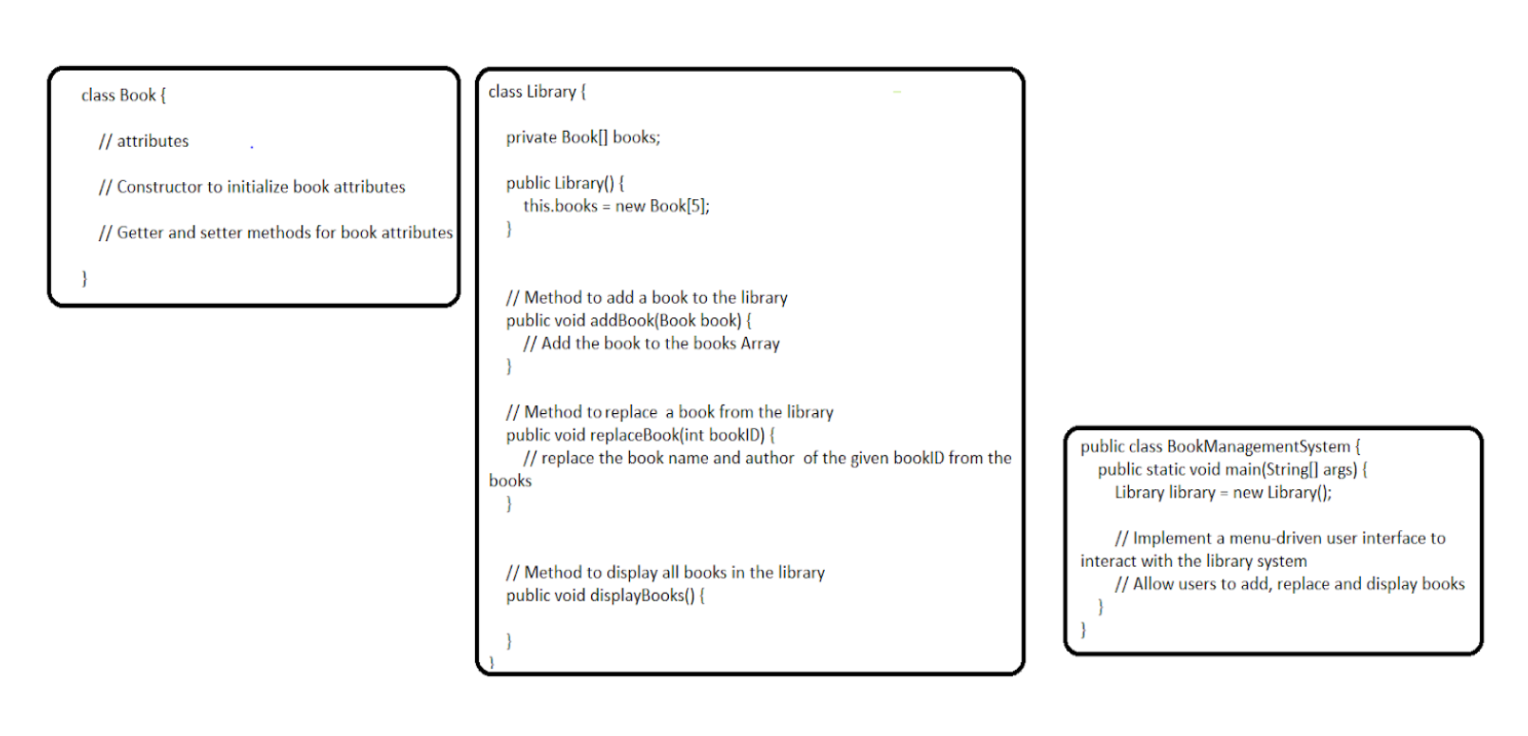
TASK - 3

Management System

1.Design a Java program that uses OOP principles to model the Book. Create two classes: Book and Library. The Book class should have attributes such as bookID, title, author, and isAvailable. The Library class should include an array to store book objects.

2.Provide methods to add books, remove book search books (using id)and display books.

Write a Java program that demonstrates the use of these classes and methods by allowing the user to interact with the library system.



Book.java

package task.three.one;

public class Book {

private int bookID;

public String title;

public String author;

public boolean isAvailable;

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

this.bookID = bookID;

this.title = title;

this.author = author;

this.isAvailable = true;

}

public int getBookID(){return bookID;}

public String getTitle(){return title;}

public String getAuthor(){return author;}

public Boolean getisAvailable(){return isAvailable;}

public void setBookID(String title){ this.title = title;}

public void setAuthor(String author){this.author = author;}

public void setisAvailable(boolean isAvailable){this.isAvailable = isAvailable;}

public String toString(){

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

}

}

Library.java

package task.three.one;

public class Library {

private Book[]books;

private int count;

public Library(){

books= new Book[5];

count = 0;

}

public void addBook(Book book){

if(count < books.length){

books[count++]=book;

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

}

else {

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

}

}

public void removeBook(int bookID){

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

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

books[i].setisAvailable(false);

System.*out*.println("Book with ID " + bookID + " marked as unavailable.");

return;

}

}

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

}

public Book searchBook(int bookID){

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

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

return books[i];

}

}

return null;

}

public void displayBooks(){

if(count == 0){

System.*out*.println("Library is empty.");

return;

}

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

System.*out*.println(books[i]);

}

}

}

BookManagement.java

package task.three.one;

import java.util.Scanner;

public class BookManagementSystem {

public static void main(String[] args) {

Library library = new Library();

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

int choice;

do {

System.*out*.println("Library 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: ");

choice = scanner.nextInt();

switch (choice) {

case 1:

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

int id = scanner.nextInt();

scanner.nextLine();

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

String title = scanner.nextLine();

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

String author = scanner.nextLine();

Book book = new Book(id, title, author);

library.addBook(book);

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 found = library.searchBook(searchID);

if (found != null) {

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

} else {

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

}

break;

case 4:

library.displayBooks();

break;

case 5:

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

break;

default:

System.*out*.println("Invalid choice. Try again.");

}

} while (choice != 5);

scanner.close();

}

}

Output -

Library Menu:

1. Add Book

2. Remove Book

3. Search Book

4. Display All Books

5. Exit

Enter your choice: 1

Enter Book ID: 1111

Enter Title: The Heaven

Enter Author: John

Book added successfully

Library Menu:

1. Add Book

2. Remove Book

3. Search Book

4. Display All Books

5. Exit

Enter your choice: 1

Enter Book ID: 2222

Enter Title: The Hell

Enter Author: Jeba

Book added successfully

Library Menu:

1. Add Book

2. Remove Book

3. Search Book

4. Display All Books

5. Exit

Enter your choice: 1

Enter Book ID: 3333

Enter Title: Doom

Enter Author: Hoe

Book added successfully

Library Menu:

1. Add Book

2. Remove Book

3. Search Book

4. Display All Books

5. Exit

Enter your choice: 1

Enter Book ID: 4444

Enter Title: Life

Enter Author: Jerry

Book added successfully

Library Menu:

1. Add Book

2. Remove Book

3. Search Book

4. Display All Books

5. Exit

Enter your choice: 1

Enter Book ID: 5555

Enter Title: Love

Enter Author: Iman

Book added successfully

Library Menu:

1. Add Book

2. Remove Book

3. Search Book

4. Display All Books

5. Exit

Enter your choice: 2

Enter Book ID to remove: 2

Book not found

Library Menu:

1. Add Book

2. Remove Book

3. Search Book

4. Display All Books

5. Exit

Enter your choice: 3

Enter Book ID to search: 4444

Book Found: BookID: 4444,Title: Life,Author: Jerry,Available: true

Library Menu:

1. Add Book

2. Remove Book

3. Search Book

4. Display All Books

5. Exit

Enter your choice: 4

BookID: 1111,Title: The Heaven ,Author: John,Available: true

BookID: 2222,Title: The Hell,Author: Jeba,Available: true

BookID: 3333,Title: Doom ,Author: Hoe,Available: true

BookID: 4444,Title: Life,Author: Jerry,Available: true

BookID: 5555,Title: Love,Author: Iman,Available: true

Library Menu:

1. Add Book

2. Remove Book

3. Search Book

4. Display All Books

5. Exit

Enter your choice: 5

Exiting

Process finished with exit code 0

2.Create Interface Taxable with members sales Tax=7% and incomeTax=10.5%. create abstract method calcTax().

a. Create class Employee(empId, name, salary) and implement Taxable to calculate income Tax on yearly salary.

b. Create class Product(pid, price, quantity) and implement Taxable to calculate sales Tax on unit price of product.

c. Create class for main method(Say DriverMain), accept employee information and a product information from user and print income tax and sales tax respectively

Taxable.java

package task.three.two;

public interface Taxable {

double *salesTax* = 0.07;

double *incomeTax* = 0.105;

void calcTax();

}

Product.java

package task.three.two;

public class Product implements Taxable{

private int pid;

private double price;

private int quantity;

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

this.pid = pid;

this.price = price;

this.quantity = quantity;

}

public void calcTax() {

double tax = price \* *salesTax*;

System.*out*.println("Sales Tax per unit for Product ID " + pid + ": ₹" + tax);

}

}

Employee.java

package task.three.two;

public class Employee implements Taxable{

private int empId;

private String name;

private double salary;

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

this.empId = empId;

this.name = name;

this.salary = salary;

}

public void calcTax() {

double yearlySalary = salary \* 12;

double tax = yearlySalary \* *incomeTax*;

System.*out*.println("Income Tax for " + name + ": ₹" + tax);

}

}

DriverMain.java

package task.three.two;

import java.util.Scanner;

public class DriverMain {

public static void main(String[] args) {

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

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

int empId = scanner.nextInt();

scanner.nextLine();

System.*out*.print("Enter Employee Name: ");

String name = scanner.nextLine();

System.*out*.print("Enter Monthly Salary: ");

double salary = scanner.nextDouble();

Employee employee = new Employee(empId, name, salary);

employee.calcTax();

System.*out*.print("\nEnter Product ID: ");

int pid = scanner.nextInt();

System.*out*.print("Enter Product Price: ");

double price = scanner.nextDouble();

System.*out*.print("Enter Product Quantity: ");

int quantity = scanner.nextInt();

Product product = new Product(pid, price, quantity);

product.calcTax();

}

}

Output-

