Question - 1

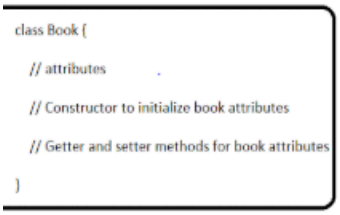
Task Descriptions :

Management System

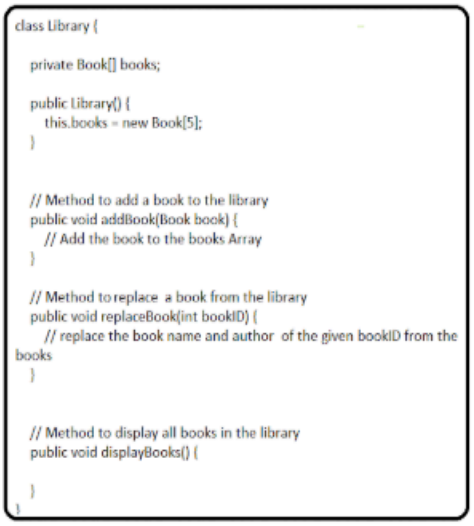
1.Design a Java program that uses OOP principles to model the Book .Create two classes: Book and Library. The Book class should have attributes such as bookID, title, author, and isAvailable. The Library class should include an array to store book objects.

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

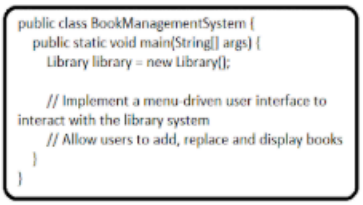
Write a Java program that demonstrates the use of these classes and methods by allowing the user to interact with the library system.



```
class Book {

    // attributes

    // Constructor to initialize book attributes

    // Getter and setter methods for book attributes

}
```



```
class Library {

    private Book[] books;

    public Library() {
        this.books = new Book[5];
    }


    // Method to add a book to the library
    public void addBook(Book book) {
        // Add the book to the books Array
    }

    // Method to replace  a book from the library
    public void replaceBook(int bookID) {
        // replace the book name and author  of the given bookID from the
books
    }


    // Method to display all books in the library
    public void displayBooks() {

    }
}
```



```
public class BookManagementSystem {
    public static void main(String[] args) {
        Library library = new Library();

        // Implement a menu-driven user interface to
interact with the library system
        // Allow users to add, replace and display books
    }
}
```

Solution

```
import java.util.Scanner;

class Book {
    private int bookID;
    private String title;
    private String author;
    private boolean isAvailable;

    public Book(int bookID, String title, String author, boolean isAvailable) {
        this.bookID = bookID;
        this.title = title;
        this.author = author;
        this.isAvailable = isAvailable;
    }

    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;
    }

    public void displayBook()
    {
        System.out.println("BookID: " + bookID + ", Title: " + title + ", Author: " + author + ",
Available: " + isAvailable);
    }
}

class Library
{
    private Book[] books;
    private int count;

    public Library()
    {
        books = new Book[10];
        count = 0;
    }

    public void addBook(Book book)
    {
        if (count < books.length)
        {
            books[count++] = book;
            System.out.println("Book added successfully.");
        }
        else
```

```
    {
        System.out.println("Library is full.");
    }
}

public void removeBook(int bookID)
{
    boolean found = false;
    for (int i = 0; i < count; i++)
    {
        if (books[i].getBookID() == bookID)
        {
            books[i] = books[count - 1];
            books[count - 1] = null;
            count--;
            found = true;
            System.out.println("Book removed.");
            break;
        }
    }
    if (!found)
    {
        System.out.println("Book not found.");
    }
}

public void searchBook(int bookID)
{
    for (int i = 0; i < count; i++)
    {
        if (books[i].getBookID() == bookID)
        {
            System.out.println("Book found:");
            books[i].displayBook();
            return;
        }
    }
    System.out.println("Book not found.");
}

public void displayBooks()
{
    if (count == 0) {
        System.out.println("No books in the library.");
    }
    else
    {
        for (int i = 0; i < count; i++) {
```

```
                books[i].displayBook();
            }
        }
    }
}

public class Main
{
    public static void main(String[] args)
    {
        Scanner scanner = new Scanner(System.in);
        Library library = new Library();

        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 (1-5): ");
            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();

                    System.out.print("Is Available (true/false): ");
                    boolean available = scanner.nextBoolean();

                    Book book = new Book(id, title, author, available);
                    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();
                library.searchBook(searchId);
                break;

            case 4:
                library.displayBooks();
                break;

            case 5:
                System.out.println("Exiting Library System. Goodbye!");
                break;

            default:
                System.out.println("Invalid choice. Please 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 (1-5): 1
Enter Book ID: 01
Enter Title: Vinland Saga
Enter Author: no one
Is Available (true/false): true
Book added successfully.

===== Library Menu =====
1. Add Book
2. Remove Book
3. Search Book
4. Display All Books
5. Exit
Enter your choice (1-5): 11
Invalid choice. Please try again.

===== Library Menu =====
1. Add Book
2. Remove Book
3. Search Book
4. Display All Books
5. Exit
Enter your choice (1-5): 1


  Enter Book ID: 02
Enter Title: AOT
Enter Author: HUMAN
Is Available (true/false): true
Book added successfully.

===== Library Menu =====
1. Add Book
2. Remove Book
3. Search Book
4. Display All Books
5. Exit
Enter your choice (1-5): 3
Enter Book ID to search: 02
Book found:
BookID: 2, Title: AOT, Author: HUMAN, Available: true
```

Question - 2

2.Create Interface Taxable with members salesTax=7% and incomeTax=10.5%. create abstract method calcTax().
a. Create class Employee(empId,name,salary) and implement Taxable to calculate incomeTax on yearly salary.
b. Create class Product(pid,price,quantity) and implement Taxable to calculate salesTax 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

```
import java.util.Scanner;
interface Taxable
{
    double salesTax = 0.07;
    double incomeTax = 0.105;

    void calcTax();
}

class Employee implements Taxable {
    int empId;
    String name;
    double salary;

    Employee(int empId, String name, double salary) {
        this.empId = empId;
        this.name = name;
        this.salary = salary;
    }

    public void calcTax() {
        double tax = salary * incomeTax;
        System.out.println("Income Tax for " + name + ": ₹" + tax);
    }
}

class Product implements Taxable {
    int pid;
    double price;
    int quantity;

    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 on Product ID " + pid + ": ₹" + tax);
    }
}

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter Employee Details:");
        System.out.print("Emp ID: ");
        int empId = sc.nextInt();
        sc.nextLine();
        System.out.print("Name: ");
        String name = sc.nextLine();
        System.out.print("Salary: ");
        double salary = sc.nextDouble();



        Employee emp = new Employee(empId, name, salary);
        emp.calcTax();
        System.out.println();
        System.out.println("Enter Product Details:");
        System.out.print("Product ID: ");
        int pid = sc.nextInt();
        System.out.print("Price: ");
        double price = sc.nextDouble();
        System.out.print("Quantity: ");
        int qty = sc.nextInt();


        Product prod = new Product(pid, price, qty);
        prod.calcTax();
    sc.close();
    }
}
```

Output

```
Enter Employee Details:
Emp ID: 27
Name: adhi
Salary: 40000
Income Tax for adhi: ₹4200.0

Enter Product Details:
Product ID: 01
Price: 2000
Quantity: 1
Sales Tax on Product ID 1: ₹140.0
```