**Lab Report**

**Course Tittle:** Object Oriented Programming Lab

**Course Code:** CSE 215

**Experiment No:** 08

**Experiment Name:** Designing Library Management System with UML & Java code

**Submitted To:**

**Name:** Mst. Umme Ayman

**Designation:** Lecturer

**Department of CSE**

**Daffodil International University**

**Submitted By**

**Name:** Md Raduan Ahamed

**ID:** 0242220005101839

**Section:** 63\_O

**Department of CSE**

**Daffodil International University**

**Submission Date:** 23-11-2023

Suppose, you are developing a LibrarySystem. Where it contains a list of books (Books), a list of librarian (Librarian), and a list of transactions (Transactions). It has methods to add books and librarian, transactions, as well as methods to handle checking out and returning books. Book has title, author, ISBN. Librarian has name, phone\_no, a list of transactions. Transaction consists of book, librarian, and date and it has a method as createTransaction which takes parameters such as book, librarian, and date to crate transaction history. LibrarySystem, Book and Librarian have the method as their name.

Question-01:Draw the UML by analyzing above scenario.

Question-02:Write java code of corresponding UML.

import java.util.ArrayList;  
import java.util.Date;  
import java.util.List;  
  
class Book {  
 private String title;  
 private String author;  
 private String ISBN;  
  
 public Book(String title, String author, String ISBN) {  
 this.title = title;  
 this.author = author;  
 this.ISBN = ISBN;  
 }  
  
 *// Getters and Setters* public String getTitle() {  
 return title;  
 }  
  
 public String getAuthor() {  
 return author;  
 }  
  
 public String getISBN() {  
 return ISBN;  
 }  
}

import java.util.ArrayList;  
import java.util.List;  
  
class Librarian {  
 private String name;  
 private String phoneNo;  
 private List<Transaction> transactions;  
  
 public Librarian(String name, String phoneNo) {  
 this.name = name;  
 this.phoneNo = phoneNo;  
 this.transactions = new ArrayList<>();  
 }  
  
 *// Getters and Setters* public String getName() {  
 return name;  
 }  
  
 public String getPhoneNo() {  
 return phoneNo;  
 }  
  
 public List<Transaction> getTransactions() {  
 return transactions;  
 }  
  
 public void addTransaction(Transaction transaction) {  
 transactions.add(transaction);  
 }  
}

import java.util.Date;  
  
class Transaction {  
 private Book book;  
 private Librarian librarian;  
 private Date date;  
  
 public Transaction(Book book, Librarian librarian, Date date) {  
 this.book = book;  
 this.librarian = librarian;  
 this.date = date;  
 }  
  
 *// Getters* public Book getBook() {  
 return book;  
 }  
  
 public Librarian getLibrarian() {  
 return librarian;  
 }  
  
 public Date getDate() {  
 return date;  
 }  
}

import java.util.ArrayList;  
import java.util.Date;  
import java.util.List;  
  
public class librarySystem {  
 private List<Book> books;  
 private List<Librarian> librarians;  
 private List<Transaction> transactions;  
  
 public librarySystem() {  
 this.books = new ArrayList<>();  
 this.librarians = new ArrayList<>();  
 this.transactions = new ArrayList<>();  
 }  
  
 *// Methods to add books and librarians* public void addBook(Book book) {  
 books.add(book);  
 }  
  
 public void addLibrarian(Librarian librarian) {  
 librarians.add(librarian);  
 }  
  
 *// Methods to handle transactions* public void checkOutBook(Book book, Librarian librarian, Date date) {  
 if (books.contains(book) && librarians.contains(librarian)) {  
 Transaction transaction = new Transaction(book, librarian, date);  
 transactions.add(transaction);  
 librarian.addTransaction(transaction);  
 System.*out*.println("Book checked out successfully.");  
 } else {  
 System.*out*.println("Book or librarian not found.");  
 }  
 }  
  
 public void returnBook(Book book, Librarian librarian, Date date) {  
 if (books.contains(book) && librarians.contains(librarian)) {  
 Transaction transaction = new Transaction(book, librarian, date);  
 transactions.add(transaction);  
 librarian.addTransaction(transaction);  
 System.*out*.println("Book returned successfully.");  
 } else {  
 System.*out*.println("Book or librarian not found.");  
 }  
 }  
  
 public static void main(String[] args) {  
 librarySystem librarySystem = new librarySystem();  
  
 Book book1 = new Book("Messege", "Dr Mizanur Rahman Azhari", "978-0-316");  
 Book book2 = new Book("English Therapy", "Saiful Islam", "978-0-06");  
  
 librarySystem.addBook(book1);  
 librarySystem.addBook(book2);  
  
 Librarian librarian = new Librarian("Raduan Ahamed", "01785566224");  
 librarySystem.addLibrarian(librarian);  
  
 Date currentDate = new Date();  
  
 librarySystem.checkOutBook(book1, librarian, currentDate);  
 librarySystem.returnBook(book2, librarian, currentDate);  
  
 *// Print transaction history for the librarian* for (Transaction transaction : librarian.getTransactions()) {  
 System.*out*.println("Transaction: Book - " + transaction.getBook().getTitle()  
 + ", Librarian - " + transaction.getLibrarian().getName()  
 + ", Date - " + transaction.getDate());  
 }  
 }  
}

**Output:**

Book checked out successfully.

Book returned successfully.

Transaction: Book - Messege, Librarian - Raduan Ahamed, Date - Sun Dec 10 16:18:07 BDT 2023

Transaction: Book - English Therapy, Librarian - Raduan Ahamed, Date - Sun Dec 10 16:18:07 BDT 2023