

```
public class Account {  
  
    private int accountID;  
    private String accountType;  
    private String fullname;  
    private String dob;  
    private String phone;  
    private String email;  
    private String address;  
    private String status;  
  
    public Account (int accountID, String accountType, String fullname, String dob, String phone, String  
email, String address, String status) {  
  
        this.accountID = accountID;  
  
        this.accountType = accountType;  
  
        this.fullname = fullname;  
  
        this.dob = dob;  
  
        this.phone = phone;  
  
        this.email = email;  
  
        this.address = address;  
  
        this.status = status;  
    }  
  
    public int getAccountID() {  
        return accountID;  
    }  
  
    public void setAccountID(int accountID) {  
        this.accountID = accountID;  
    }  
  
    public String getAccountType() {
```

```
        return accountType;
    }

    public void setAccountType(String accountType) {
        this.accountType = accountType;
    }

    public String getFullName() {
        return fullname;
    }

    public void setFullName(String fullname) {
        this.fullname = fullname;
    }

    public String getDob() {
        return dob;
    }

    public void setDob(String dob) {
        this.dob = dob;
    }

    public String getPhone() {
        return phone;
    }

    public void setPhone(String phone) {
        this.phone = phone;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }
}
```

```

    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }

    public String getStatus() {
        return status;
    }

    public void setStatus(String status) {
        this.status = status;
    }
}

public class Book{
    private int ISBN;
    private String title;
    private String authors;
    private int year;
    private String publisher;
    private String status;

    public Book(int ISBN, String title, String authors, int year, String publisher, String status){
        this.ISBN = ISBN;
        this.title = title;
        this.authors = authors;
    }
}

```

```
        this.year= year;

        this.publisher = publisher;

        this.status = status;
    }

    public int getISBN() {
        return ISBN;
    }


    public void setISBN(int ISBN) {
        ISBN = ISBN;
    }


    public String getTitle() {
        return title;
    }


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


    public String getAuthors() {
        return authors;
    }


    public void setAuthors(String authors) {
        this.authors = authors;
    }


    public int getYear() {
```

```
    return year;
}
```

```
public void setYear(int year) {
    this.year = year;
}
```

```
public String getPublisher() {
    return publisher;
}
```

```
public void setPublisher(String publisher) {
    this.publisher = publisher;
}
```

```
public String getStatus() {
    return status;
}
```

```
public void setStatus(String status) {
    this.status = status;
}
}
```

```
public class BookRequest {
    private int BLcode;
    private int memberID;
    private int ISBN;
```

```
private String dueDate;
```

```
private String returnedDate;
```

```
public BookRequest(){};
```

```
public BookRequest(int BLcode, int memberID, int ISBN, String dueDate, String returnedDate) {
```

```
    this.BLcode = BLcode;
```

```
    this.memberID = memberID;
```

```
    this.ISBN = ISBN;
```

```
    this.dueDate = dueDate;
```

```
    this.returnedDate = returnedDate;
```

```
}
```

```
public int getBLcode() {
```

```
    return BLcode;
```

```
}
```

```
public void setBLcode(int bLcode) {
```

```
    BLcode = bLcode;
```

```
}
```

```
public int getMemberID() {
```

```
    return memberID;
```

```
}
```

```
public void setMemberID(int memberID) {
```

```
    this.memberID = memberID;
```

```
}
```

```
public int getISBN() {
```

```
    return ISBN;
```

```
}
```

```
public void setISBN(int iISBN) {
```

```

        ISBN = iISBN;
    }
    public String getDueDate() {
        return dueDate;
    }
    public void setDueDate(String dueDate) {
        this.dueDate = dueDate;
    }
    public String getReturnedDate() {
        return returnedDate;
    }
    public void setReturnedDate(String returnedDate) {
        this.returnedDate = returnedDate;
    }
}

```

```
import java.util.List;
```

```

public class LibrarianAccount extends Account{
    private String startingDate;
    private String married;

    public LibrarianAccount(int accountID, String accountType, String fullname, String dob, String
phone,String email, String address ,String status,String startingDate, String married){
        super(accountID, accountType, fullname, dob, phone,email,address, status);
        this.startingDate = startingDate;
    }
}

```

```

        this.married = married;
    }

    public String getStartingDate() {
        return startingDate;
    }

    public void setStartingDate(String startingDate) {
        this.startingDate = startingDate;
    }

    public String getMarried() {
        return married;
    }

    public void setMarried(String married) {
        this.married = married;
    }

    // libraian find book

    public void findBookByISBN (LibraryManagementSystem manager , int id){
        List<Book> find = manager.findBook(id);
        if (find.isEmpty()){
            System.out.println("Khon tim thay sach dua theo ISBN.");
        } else{
            System.out.println("Da tim thay sach:");
            for (Book book : find){
                System.out.println(book.getTitle());
            }
            System.out.println("Nhan vien tim thay");
        }
    }

```



```

}

// librarian find book by name

public void findMemberName(LibraryManagementSystem manager, String memberName){

    List<MemberAccount> members = manager.searchMemberName(memberName);

    if (members.isEmpty()){

        System.out.println("Khong thay thanh vien nao theo ten da cho.");

    }else {

        System.out.println("Tim thay thanh vien:");

        for (MemberAccount m : members ){

            System.out.println(m.getFullname());

        }

        System.out.println("Nhan vien tim thay ");

    }

}

// librarian find member by id account

public void findMemberById(LibraryManagementSystem manager, int id){

    List<MemberAccount> members = manager.searchMemberID(id);

    if (members.isEmpty()){

        System.out.println("Khong thay thanh vien nao theo ID da cho.");

    }else {

        System.out.println ("Tim thay thanh vien: ");

        for (MemberAccount m : members){

            System.out.println(m.getFullname());

        }

        System.out.println("Nhan vien tim thay");

    }

}

public void findBookRequests(LibraryManagementSystem manager, int number) {

    List<BookRequest> bookRequests = manager.searchBookRequests(number);

```

```

    if (bookRequests.isEmpty()) {
        System.out.println("Khong thay phieu yeu cau theo tu khoa da cho");
    } else {
        System.out.println("Tim thay phieu yeu cau:");
        for (BookRequest request : bookRequests) {
            System.out.println("BLcode: " + request.getBLcode() +
                ", Member ID: " + request.getMemberID() +
                ", ISBN: " + request.getISBN() +
                ", Due Date: " + request.getDueDate());
        }
    }
}

```

```

import java.util.ArrayList;
import java.util.List;

```

```

public class LibraryManagementSystem {
    private List <Book> catalog;
    private List <MemberAccount> members;
    private List <BookRequest> requests;

    public LibraryManagementSystem() {
        this.catalog = new ArrayList<>();
        this.members = new ArrayList<>();
        this.requests = new ArrayList<>();
    }
}

```

```

    }

    public List<Book> getCatalog() {
        return catalog;
    }

    public void setCatalog(List<Book> catalog) {
        this.catalog = catalog;
    }

    // method find book by id
    public List<Book> findBook (int bookId){
        List<Book> id = new ArrayList<>(); // id cua book

        // duyet
        for (Book book : catalog){
            // neu tim thay
            if (book.getISBN()== bookId){
                // add book
                id.add(book);
            }
        }

        // tra ve
        return id;
    }

    public List<Book> searchBookKeword(String keyword){
        List<Book> searchbook = new ArrayList<>();

        for (Book book : catalog){
            if (book.getAuthors().equals(keyword) || book.getPublisher().equals(keyword) ||
book.getTitle().equals(keyword)){
                searchbook.add(book);
            }
        }
    }

```

```
    }  
    return searchbook;  
}
```

```
// method find member name
```

```
public List<MemberAccount> searchMemberName (String keyWord){  
    List<MemberAccount> member = new ArrayList<>();  
    for (MemberAccount m : members){  
        if (m.getFullname().equals(keyWord)){  
            member.add(m);  
        }  
    }  
    return member;  
}
```

```
// method find member id
```

```
public List<MemberAccount> searchMemberID (int id){  
    List<MemberAccount> member = new ArrayList<>();  
    for (MemberAccount m : members){  
        if (m.getAccountID() == id){  
            member.add(m);  
        }  
    }  
    return member;  
}
```

```
public void createBookRequest (BookRequest request){  
    requests.add(request);  
}
```

```

        System.out.println("Phieu muon sach da duoc tao");
    }

    // Hiển thị danh sách sách đã được mượn
    public void displayBorrowedBooks() {
        System.out.println("Cac sach da cho muon:");
        for (Book book : catalog) {
            if (book.getStatus().equals("da muon")) {
                System.out.println("ISBN: " + book.getISBN() + ", De muc: " + book.getTitle());
            }
        }
    }

    /// Hiển thị danh sách sách có sẵn
    public void displayAvailableBooks() {
        System.out.println("Sach co san:");
        for (Book book : catalog) {
            if (book.getStatus().equals("co san")) {
                System.out.println("ISBN: " + book.getISBN() + ", De muc: " + book.getTitle());
            }
        }
    }

    // cap nhat trang thai của sách
    public void updateBookStatus(int ISBN, String newStatus) {
        for (Book book : catalog) {
            if (book.getISBN() == ISBN) {
                book.setStatus(newStatus);
                break;
            }
        }
    }
}

```

```

public List<BookRequest> searchBookRequests(int number) {
    List<BookRequest> searchRequests = new ArrayList<>();

    for (BookRequest request : requests) {
        // Tìm kiếm theo mã mượn
        if (number == request.getBLcode()) {
            searchRequests.add(request);
        }
    }

    return searchRequests;
}
}

```

```

import java.util.ArrayList;
import java.util.List;

```

```

public class Main {
    public static void main(String[] args) {
        LibraryManagementSystem librarySystem = new LibraryManagementSystem();

        // Khởi tạo danh mục sách
        List<Book> catalog = new ArrayList<>();
        librarySystem.setCatalog(catalog);

        // Thêm sách vào danh mục
    }
}

```

```

Book book1 = new Book(1, "Introduction to Java", "John Doe", 2022, "Publisher A", "co san");

Book book2 = new Book(2, "Data Structures in Python", "Jane Smith", 2021, "Publisher B", "co
san");

Book book3 = new Book( 3,"Neu nhu chi con 1 ngay de song","An",2020, "NXB tre", "da muon");

catalog.add(book1);

catalog.add(book2);

catalog.add(book3);


// Hiển thị danh sách sách
System.out.println("Danh muc thu vien:");

for (Book book : catalog) {
    System.out.println("De muc: " + book.getTitle() + ", Tinh trang: " + book.getStatus());
}

// hien thi danh sách sách có sẵn và sách ko có sẵn
librarySystem.displayAvailableBooks();

librarySystem.displayBorrowedBooks();


// tạo một số nhân viên thư viện
LibrarianAccount librarian1 = new LibrarianAccount (1,"L","Ngoc","2004","09123","dat@", "hcm",
"active","2021-9-9 ","married");

// Tạo một số thành viên thư viện
MemberAccount member1 = new MemberAccount(101, "M", "An", "2000-01-01", "123456789",
"nhi@email.com", "123 thu duc", "active", "2022-01-01", 2);

MemberAccount member2 = new MemberAccount(102, "M", "Khang", "1995-05-15", "987654321",
"trung@email.com", "456 quan 9", "active", "2022-02-15", 1);


// librarian find book
librarian1.findBookByISBN(librarySystem, 3);


// Hiển thị thông tin thành viên

```

```

        System.out.println("\nThanh vien thu vien:");

        System.out.println("Thanh vien 1: " + member1.getFullname() + ", thoi han thanh vien: " +
member1.getMembershipDate());

        System.out.println("Thanh vien 2: " + member2.getFullname() + ", thoi han thanh vien: " +
member2.getMembershipDate());


        // tim sach

        member1.findBookISBN(librarySystem, 2);

        // Thành viên 1 mượn sách

        member1.requestBook(librarySystem);


        librarian1.findBookRequests(librarySystem, 1);

    }
}

```

```

import java.util.List;

```

```

import java.util.Scanner;

```

```

public class MemberAccount extends Account {

```

```

    Scanner input = new Scanner(System.in);

```

```

    private String membershipDate;

```

```

    private int nBooks;

```

```

    public MemberAccount (int accountID, String accountType,String fullname,
        String dob, String phone, String email, String address, String status, String membershipDate,
        int nBooks) {

```



```

    super (accountID, accountType, fullname, dob, phone,email,address, status);

    this.nBooks = nBooks;

    this.membershipDate = membershipDate;
}

public String getMembershipDate() {
    return membershipDate;
}

public void setMembershipDate(String membershipDate) {
    this.membershipDate = membershipDate;
}

public int getnBooks() {
    return nBooks;
}

public void setnBooks(int nBooks) {
    this.nBooks = nBooks;
}

// find by id
public void findBookISBN (LibraryManagementSystem manager, int id){
    List<Book> find = manager.findBook(id);
    if (find.isEmpty()){
        System.out.println("Khong tim thay sach dua theo ISBN .");
    }else{
        System.out.println("Da tim thay sach:");
        for (Book book : find){
            System.out.println(book.getTitle());
        }
    }
}

```

```

        System.out.println("Thanh cong");
    }
}

// find by others
public void findBook (LibraryManagementSystem manager, String keyword){
    List<Book> find = manager.searchBookKeword(keyword);
    if (find.isEmpty()){
        System.out.println("Khong tim thay sach dua theo tu khoa.");
    }else {
        System.out.println("Da tim thay sach:");
        for (Book book : find){
            System.out.println(book.getTitle());
        }
        System.out.println("Thanh cong");
    }
}

public void requestBook(LibraryManagementSystem manager){
    // Hiển thị danh sách sách trong catalog để thành viên chọn
    System.out.println("Sach co san trong danh muc:");
    manager.displayAvailableBooks();

    // Nhập ISBN của sách cần mượn từ thành viên
    System.out.print("Nhap ISBN cua cuon sach ban muon: ");
    int requestedISBN = input.nextInt();

    // Nhập thông tin cho BookRequest từ thành viên
    System.out.print("Nhap BLcode: ");
    int newBLcode = input.nextInt();
    input.nextLine(); // Consume the newline character
}

```

```

        System.out.print("Nhap ngay den han (yyyy-MM-dd): ");

        String newdueDate = input.nextLine();

        // Create a BookRequest object
        BookRequest request = new BookRequest(newBLcode, this.getAccountID(), requestedISBN,
newdueDate,null);

        manager.createBookRequest(request);

        System.out.print("Muon thanh cong!! ");

        this.updateBookStatus(manager,requestedISBN,"da muon");
    }

    // Cập nhật số sách đã mượn
    public void updateNBooks(int increment) {
        this.nBooks += increment;
    }

    // Cập nhật trạng thái của sách
    public void updateBookStatus(LibraryManagementSystem manager, int ISBN, String newStatus) {
        manager.updateBookStatus(ISBN, newStatus);
    }
}

```