Name - Rohit Yadav
Roll No. - 2401730144
B.Tech CSE (AI/ML) Section - B

21/11/2025

Assignment - 4

```java
import java.io.*;
import java.util.*;

interface Show {void show();}

abstract class Item implements Show {
    int id; String title;
    Item(id, String title) { this.id = id; this.title = title; }
}

class Book extends Item {
    String auth, cat; boolean issued;
    Book(int id, String t, String a, String c) { super(id, t);
        auth = a; cat = c; }
    void issue() { issued = true; }
    void ret() { issued = false; }
    public void show() { System.out.println(id + "|" + title +
        "|" + auth + "|" + cat + "|" + issued); }
}

class Member implements Show {
    int mid; String name, email; List <Integer> list = new
            ArrayList <>();
    Member(int id, String n, String c) { mid = id; name = n;
        email = e; }
    void add(int id) { list.add(id); }
    void rem(int id) { list.remove(Integer.valueOf(id)); }
    public void show() { System.out.println(mid + "|" +
        name + "|" + email + "|" + list); }
}

class BookErr Extends Exception {
    BookErr(Stringm) { super(m); }
}
```

```java
class Lib{
    Map< Integer, Book> bmap = new Hash Map<>();
    Map< Integer, Member> mmap = new Has Map<>();
    int bc = 100, mc= 200;
    Lib (){ load(); auto();}
    void addBook (String t, String a, String c){
        Book b = new Book(++bc, t, a, c);
        bmap.put (b.id, b);
        System.out.println("Book ID: " +b.id );
    }

    void addMem(String n, String e){
        Member m = new Member(++mc, n, e);
        mmap.put (m.mid, m);
        System.out.println("Member ID: "+ m.mid);
    }

    void issue (int bid, int mid) throws BookErr{
        if (! bmap.contains Key (bid)||! mmap.containsKey
        (mid)) return;
        Book b = bmap.get (bid);
        if (b.issued) throw new Book Err("Issued");
        b.issue();
        mmap.get (mid).add(bid);
        System.out.println("Done");
    }

    void ret (int bid, int mid){
        if(! bmap.contains Key (bid)||! mmap.containsKey(mid))
        return;
        bmap.get (bid).ret();
        mmap.get (mid).rem(bid);
        System.out.println("Returned");
    }
    void Search (String k){
```

```java
bmap.values().stream().filter(b -> b.title.contain(k)
    || b.auth.contains(k) || b.cat.contains(k)). for Each
    (Book i: show);
}

void sort(){
    bmap.values().stream().sorted(Comparator.compari
    -ng(b -> b.title)). for Each(Book :: show);
}

void save(){
try (BufferedWriter w = new BufferedWriter(new File
    Writer("books.txt"))){
    for(Book b: bmap.values()) w. write (b.id+", "+b.
        title+", "+b.auth+", "+b.cat+", "+b.issued
        + "\n");
} catch (Exception e){}
try (BufferedWriter w = newBufferedWriter(new
    FileWriter("members.txt"))){
    for (member m: map.values()) w.write (m.mid+","
    + m.name+", "+m.email+", "+m.list +"\n");
} catch(Exception e){}
}

void load (){
try (BufferedReader r = new BufferedReader(newFile
Reader("books.txt"))){
    Strings; while ((s = r.read Line()) != null){
        String p[] = s.split(", ");
        Book b = new Book (Integer.parse Int(p[0]), p[1], p[2],
        p[3]);
        b.issued = Boolean.parse Boolean(p[4]);
        bmap.put(b.id, b); bc = Math.max(bc. b.id);
    }
} catch (Exception e){}
```

```java
try(BufferedReader r = new BufferedReader (new File
Reader ("members.txt"))) {
    String s; while((s= r.readLine())!= null){
    String p[] = s. split (", ");
Member m = new Member( Integer.parseInt(p[0], p[1],
    p[2]);
mmap.put (m.mid,m); mc=Math.max (mc, m.mid);
    }
} catch (Exception e) {}
}

void auto () {
    Thread t=new Thread (()->{try {while(true) {save();
    Thread.sleep (3000);}} catch (Exceptione ){}});
    t. setDaemon(true) ; start();
}
}

public class Library System {
    public static void main (String []a){
    Lib l =new Lib();
Scanner s= new Scanner (System.in);
while (true) {
    System.out. println("1 Add Book2 Add Mem 3 Issue 4Return
        5 Search 6 Sort.7 Exit");
    try {
    int c= s.nextInt();
    switch (c) {
    Case 1 ->{
    s.nextLine ();
    Sout ("Title?"); String t=s.nextLine();
    Sout (" Auth: "); String au= s.nextLine ();
    Sout ("Cat: "); String c1 = s.nextLine();
    l.add Book(t,au, c1);
}
}
```

```java
case 2 -> {
  s.nextLine();
  Sout(" Name:"); String n = s.nextLine();
  Sout(" Email:"); String e = s.nextLine();
  l.addMem(n, e);
}

case 3 -> {
  System.out.print("Bid:"); int bid = s.nextInt();
  System.out.print("Mid:"); int mid = s.nextInt();
  l.issue(bid, mid);
}

case 4 -> {
  Sout("Bid:"); int bid = s.nextInt();
  Sout("Mid:"); int mid = s.nextInt();
  l.ret(bid, mid);
}

case 5 -> {
  s.nextLine();
  Sout("Key:"); l.search(s.nextLine());
}

Case 6 -> l.sort();
Case 7 -> { l.save(); return; }
}
} catch (BookErr e) { System.out.print(e.getMessage()); }
  Catch (Exception e) { System.out.println("Err"); s.
  nextLine(); }
  }
}
}
```