JUGNAR SINGH
2401730108

Page No.
Date

## JAVA ASSIGNMENT 4

```
import java.io.*;
import java.util.*;
class Book implements
Comparable <Book> {
  int id; String t, a, c; boolean issued;
  Book(int i, String t, String, a, String c)
  { id = i; this.t = t; this.a = a; this.c=c;}
  public int compareTo (Book o) { return t.
  compareToIngnoreCase (o.t); }
}
class Member {
  int id; String n, e; List< Integer>
list = new ArrayList<>();
  Member(int i, String n, String e)
  {id = i; this.n = n; this.e = e; }
}
public class Library {
  static Map< Integer, Book> books = new HashMap<>();
  static Map < Integer, Member>
members = new HashMap <>();
  static File fb = new File ("books.txt");
fm = new File ("members.txt");
   public static void main (String [] z) {
     load ();
     Scanner sc = new Scanner (System.in);
       while (true) {
         System.out.println ("\n1. Add Book
         2. Add Member 3. Issue 4. Return
         5. Search 6. Sort 7. Exit ");
```

```
        int ch = sc.nextInt(); sc.~~ne~~nextLine();
        if (ch==1) addBook(sc);
        else if (ch==2) addMember(sc);
        else if (ch==3) issue(sc);
        else if (ch==4) returnBook(sc);
        else if (ch==5) search(sc);
        else if (ch==6) sort();
        else System.exit();
      }
   }

   static void addBook(Scanner s){
        System.out.print("ID: "); int id =
        s.nextInt(); s.nextLine();
        System.out.print("Title: "); String t = s.nextL
        System.out.print("category: "); String c = s.nextLine
        books.put(id, new Book(id, t, a, c));
        save();
   }

   static void addMember(Scanner s) {
        System.out.print("ID: "); int id= s.nextInt()
        s.nextLine();
        System.out.print("Name: "); String n = s.nextL
        System.out.print("Email: "); String e = s.nextL
        members.put(id, new Member(id, n, e));
        save();
   }
```

```
static void issue (Scanner s) {
    System.out.print ("BOOK ID : "); int b =
    s.nextInt ();
    System.out.print ("Member ID : "); int
    m=s.nextIn();
    ~~if (books.get (b). issued) r~~
    if (!books. containsKey (b)) ||! members.
    containsKey (m)) return;
    if (books.get (b). issue = true;
    members. get (m). list.add (b);

    save ();
  }
              returnBook
 static void ~~[illegible]~~ ( ) {
     System.out.print ("BOOK ID : "); int b =
     s.nextInt ();
     if(! books. containsKey (b)) return ;
     books. get (b). issued = false;
     members. values (). forEach (x → x. list.
  remove ((Integer) b));
     save ();
 }
 static void search (Scamer s) {
     System. out. print ("Search : "); String
 K = s.nextLine(). tolowercase ();
     books.values (). steam (). filter (x → x. t.
     to. LowerCase (). Contains (K) ||. x. a. to
     LowerCase (). ~~[illegible]~~ contains (K) || x. c. to
     LowerCase (). Contains (K)). forEach ( x →
     System. out. println (x. id + " " + x. t));
 }
```

```
static void sort(){
  List<Book> l = new ArrayList<>(books.values(
  System.out.println("1. Title  2. Author");
  Scanner s = new Scanner(System.in);
  int c = s.nextInt();
  if(c==1) Collections.sort(l);
  else l.sort((x,y) -> x.a.compareToIgnoreCase
  (y.a));
  l..forEach(x -> System.out.println(x.id + " "+
  x.t + " " + x.a));
}

static void save(){
  try(PrintWriter pw = new PrintWriter(fb)){
    for(Book b: books.values())
        pw.println(b.id+","+b.t+","+b.at",
        + b.c+","+ b.issued);
    } catch(Exception e){}
    try(PrintWriter pw = new PrintWriter(fm)
    { for(Member m: members.values())
         pw.println(m.id+","+ m.n +","+
    m.e+","+ m.list);
      }catch(Exception e){}
}
static void load(){
    try (BufferedReader br = new Buffered
    Reader (new FileReader (fb))){ String I; while
    ((I = br.readLine ())! = null){
        String a[] = I.split (",");
        Book b = new Book(Integer.parseInt
        (a[0]), a[1], a[2], a[3]);
        b.issued = Boolean.parseBoolean (a[4]);
                books.put(b.id, b);
        }
    } catch(Exception e) {}
    try (BufferedReader br = new Buffered
    Reader(new FileReader (fm))){ String I;
      while((I = br.readLine()), != null){
          ~~String a[] =~~
        String a[] = I.split(",", 4);
        Member m = new Member (Integer.parseInt
        (a[0]), a[1], a[2] );
        if(a.length == 4){
            String s = a[3]. replace("[", "")
      .replace("]", "");
            if(!s.isEmpty())
    for (String x : s.split(","))

  m.list.add(Integer.parseInt(x.trim()));
        }
        members.put(m.id, m);
    }
  }catch(Exception e) {}
  }
}
```