



K.R. Mangalam University  
School of Engineering & Technology

Fundamentals Of Java Programming  
(ENCA203) Assignment 4  
City Library Digital Management System.

Submitted by:

Name: RAKESH G  
Kumar

Roll No: 2401201064

Class: BCA (AI & DS)

Submitted to:

Dr. Manish

GitHub Repository:

[https://github.com/rakesh4407/Java\\_Assignment\\_Rakesh](https://github.com/rakesh4407/Java_Assignment_Rakesh)

# Code:

```
Rakesh_Assignment.ipynb  ResultManager.java 7  LibrarySystem.java 9+ X
JAVA > LibrarySystem.java > Language Support for Java(TM) by Red Hat > LibrarySystem > Book > getBookId()
1
2 import java.io.*;
3 import java.util.*;
4
5 public class LibrarySystem {
6
7     static class Book implements Comparable<Book> {
8
9         private int bookId;    Field bookId can be final
10        private String title;    Field title can be final
11        private String author;    Field author can be final
12        private String category;    Field category can be final
13        private boolean isIssued;
14
15        public Book(int bookId, String title, String author, String category) {
16            this.bookId = bookId;
17            this.title = title;
18            this.author = author;
19            this.category = category;
20            this.isIssued = false;
21        }
22
23        public int getBookId() {
24            return bookId;
25        }
26
27        public String getTitle() {
28            return title;
29        }
30
31        public String getAuthor() {
32            return author;
33        }
34
35        public String getCategory() {
36            return category;
37        }
38
39        public boolean isIssued() {
40            return isIssued;
41        }
42
43        public void markAsIssued() {
44            isIssued = true;
45        }
46
47        public void markAsReturned() {
48            isIssued = false;
49        }
50
51        public void displayBookDetails() {
52            System.out.println("Book ID: " + bookId);
53            System.out.println("Title: " + title);
54            System.out.println("Author: " + author);
55            System.out.println("Category: " + category);
56            System.out.println("Issued: " + (isIssued ? "Yes" : "No"));
57        }
58
59        @Override
60        public int compareTo(Book other) {
61            return this.title.compareToIgnoreCase(other.title);
62        }
63
64        @Override
65        public String toString() {
66            return bookId + "," + title + "," + author + "," + category + "," + isIssued;
67        }
68
69        public static Book fromString(String line) {
70            String[] p = line.split(",");
71            if (p.length == 5) {
72                Book b = new Book(Integer.parseInt(p[0]), p[1], p[2], p[3]);
73                b.isIssued = Boolean.parseBoolean(p[4]);
74                return b;
75            }
76            return null;
77        }
78    }
79
80    static class Member {
81
82        private int memberId;    Field memberId can be final
83        private String name;    Field name can be final
84        private String email;    Field email can be final
85        private List<Integer> issuedBooks = new ArrayList<>();    Field issuedBooks can be final
86
87        public Member(int memberId, String name, String email) {
88            this.memberId = memberId;
89            this.name = name;
90            this.email = email;
```

```

Rakesh_Assignment.ipynb  ResultManager.java 7  LibrarySystem.java 9+ X
JAVA > LibrarySystem.java > Language Support for Java(TM) by Red Hat > LibrarySystem > Book > getBookId()
5 public class LibrarySystem {
80     static class Member {
87         public Member(int memberId, String name, String email) {
91             }
92
93         public int getMemberId() {
94             return memberId;
95         }
96
97         public String getName() {
98             return name;
99         }
100
101         public String getEmail() {
102             return email;
103         }
104
105         public void addIssuedBook(int bookId) {
106             issuedBooks.add(bookId);
107         }
108
109         public void returnIssuedBook(int bookId) {
110             issuedBooks.remove(Integer.valueOf(bookId));
111         }
112
113         public void displayMemberDetails() {
114             System.out.println("Member ID: " + memberId);
115             System.out.println("Name: " + name);
116             System.out.println("Email: " + email);
117             System.out.println("Issued Books: " + issuedBooks);
118         }
119
120         @Override
121         public String toString() {
122             StringBuilder sb = new StringBuilder();
123             sb.append(memberId).append(",").append(name).append(",").append(email).append(",");
124             for (int i = 0; i < issuedBooks.size(); i++) {
125                 sb.append(issuedBooks.get(i));
126                 if (i < issuedBooks.size() - 1) {
127                     sb.append(",");
128                 }
129             }
130             return sb.toString();
131         }
132
133         public static Member fromString(String line) {

```

```

Rakesh_Assignment.ipynb  ResultManager.java 7  LibrarySystem.java 9+ X
JAVA > LibrarySystem.java > Language Support for Java(TM) by Red Hat > LibrarySystem > Book > getBookId()
5 public class LibrarySystem {
80     static class Member {
133         public static Member fromString(String line) {
134             String[] p = line.split(",", 4);
135             if (p.length >= 3) {
136                 Member m = new Member(Integer.parseInt(p[0]), p[1], p[2]);
137                 if (p.length == 4 && !p[3].isEmpty()) {
138                     String[] books = p[3].split(";");
139                     for (String b : books) {
140                         m.issuedBooks.add(Integer.parseInt(b));  Unnecessary temporary when converting from String
141                     }
142                 }
143                 return m;
144             }
145             return null;
146         }
147     }
148
149     static class LibraryManager {
150
151         private Map<Integer, Book> books = new HashMap<>();  Field books can be final
152         private Map<Integer, Member> members = new HashMap<>();  Field members can be final
153         private final String BOOK_FILE = "books.txt";
154         private final String MEMBER_FILE = "members.txt";
155         private Scanner sc = new Scanner(System.in);  Field sc can be final
156
157         public LibraryManager() {
158             loadFromFile();
159         }
160
161         private void saveBooks() {
162             try (BufferedWriter bw = new BufferedWriter(new FileWriter(BOOK_FILE))) {
163                 for (Book b : books.values()) {
164                     bw.write(b.toString());
165                     bw.newLine();
166                 }
167             } catch (IOException e) {
168                 System.out.println("Error saving books: " + e.getMessage());
169             }
170         }
171
172         private void saveMembers() {
173             try (BufferedWriter bw = new BufferedWriter(new FileWriter(MEMBER_FILE))) {
174                 for (Member m : members.values()) {
175                     bw.write(m.toString());
176                     bw.newLine();

```

```

Rakesh_Assingment.ipynb  ResultManager.java 7  LibrarySystem.java 9+ X
JAVA > LibrarySystem.java > Language Support for Java(TM) by Red Hat > LibrarySystem > Book > getBookId()
5 public class LibrarySystem {
149     static class LibraryManager {
172         private void saveMembers() {
178             } catch (IOException e) {
179                 System.out.println("Error saving members: " + e.getMessage());
180             }
181         }
182
183         private void saveToFile() {
184             saveBooks();
185             saveMembers();
186         }
187
188         private void loadFromFile() {
189             try (BufferedReader br = new BufferedReader(new FileReader(BOOK_FILE))) {
190                 String line;
191                 while ((line = br.readLine()) != null) {
192                     Book b = Book.fromString(line);
193                     if (b != null) {
194                         books.put(b.getBookId(), b);
195                     }
196                 }
197             } catch (IOException e) {
198             }
199
200             try (BufferedReader br = new BufferedReader(new FileReader(MEMBER_FILE))) {
201                 String line;
202                 while ((line = br.readLine()) != null) {
203                     Member m = Member.fromString(line);
204                     if (m != null) {
205                         members.put(m.getMemberId(), m);
206                     }
207                 }
208             } catch (IOException e) {
209             }
210         }
211
212         public void addBook() {
213             try {
214                 System.out.print("Enter Book ID: ");
215                 int id = sc.nextInt();
216                 sc.nextLine();
217                 if (books.containsKey(id)) {
218                     System.out.println("Book ID already exists.");
219                     return;
220                 }

```

```

Rakesh_Assingment.ipynb  ResultManager.java 7  LibrarySystem.java 9+ X
JAVA > LibrarySystem.java > Language Support for Java(TM) by Red Hat > LibrarySystem > Book > getBookId()
5 public class LibrarySystem {
149     static class LibraryManager {
212         public void addBook() {
221             System.out.print("Enter Title: ");
222             String title = sc.nextLine();
223             System.out.print("Enter Author: ");
224             String author = sc.nextLine();
225             System.out.print("Enter Category: ");
226             String category = sc.nextLine();
227
228             Book b = new Book(id, title, author, category);
229             books.put(id, b);
230             saveBooks();
231             System.out.println("Book added successfully!");
232         } catch (Exception e) {
233             System.out.println("Error adding book: " + e.getMessage());
234         }
235     }
236
237     public void addMember() {
238         try {
239             System.out.print("Enter Member ID: ");
240             int id = sc.nextInt();
241             sc.nextLine();
242             if (members.containsKey(id)) {
243                 System.out.println("Member ID already exists.");
244                 return;
245             }
246             System.out.print("Enter Name: ");
247             String name = sc.nextLine();
248             System.out.print("Enter Email: ");
249             String email = sc.nextLine();
250
251             Member m = new Member(id, name, email);
252             members.put(id, m);
253             saveMembers();
254             System.out.println("Member added successfully!");
255         } catch (Exception e) {
256             System.out.println("Error adding member: " + e.getMessage());
257         }
258     }
259
260     public void issueBook() {
261         try {
262             System.out.print("Enter Member ID: ");
263             int mid = sc.nextInt();

```

```

Rakesh_Assignment.ipynb  ResultManager.java 7  LibrarySystem.java 9+ X
JAVA > LibrarySystem.java > Language Support for Java(TM) by Red Hat > LibrarySystem > Book > getBookId()
5 public class LibrarySystem {
149     static class LibraryManager {
260     public void issueBook() {
264         System.out.print("Enter Book ID: ");
265         int bid = sc.nextInt();
266
267         Book b = books.get(bid);
268         Member m = members.get(mid);
269
270         if (b == null) {
271             System.out.println("Book not found.");
272             return;
273         }
274         if (m == null) {
275             System.out.println("Member not found.");
276             return;
277         }
278         if (b.isIssued()) {
279             System.out.println("Book already issued.");
280             return;
281         }
282
283         b.markAsIssued();
284         m.addIssuedBook(bid);
285         saveToFile();
286         System.out.println("Book issued successfully!");
287     } catch (Exception e) {
288         System.out.println("Error issuing book: " + e.getMessage());
289     }
290 }
291
292 public void returnBook() {
293     try {
294         System.out.print("Enter Member ID: ");
295         int mid = sc.nextInt();
296         System.out.print("Enter Book ID: ");
297         int bid = sc.nextInt();
298
299         Book b = books.get(bid);
300         Member m = members.get(mid);
301
302         if (b == null || m == null) {
303             System.out.println("Invalid IDs.");
304             return;
305         }
306         if (!b.isIssued()) {

```

```

Rakesh_Assignment.ipynb  ResultManager.java 7  LibrarySystem.java 9+ X
JAVA > LibrarySystem.java > Language Support for Java(TM) by Red Hat > LibrarySystem > Book > getBookId()
5 public class LibrarySystem {
149     static class LibraryManager {
292     public void returnBook() {
308         return;
309     }
310
311     b.markAsReturned();
312     m.returnIssuedBook(bid);
313     saveToFile();
314     System.out.println("Book returned successfully!");
315 } catch (Exception e) {
316     System.out.println("Error returning book: " + e.getMessage());
317 }
318 }
319
320 public void searchBooks() {
321     sc.nextLine();
322     System.out.print("Enter keyword (title/author/category): ");
323     String key = sc.nextLine().toLowerCase();
324     for (Book b : books.values()) {
325         if (b.getTitle().toLowerCase().contains(key)
326             || b.getAuthor().toLowerCase().contains(key)
327             || b.getCategory().toLowerCase().contains(key)) {
328             b.displayBookDetails();
329             System.out.println("-----");
330         }
331     }
332 }
333
334 public void sortBooks() {
335     System.out.println("1. Sort by Title\n2. Sort by Author");
336     int ch = sc.nextInt();
337     List<Book> list = new ArrayList<>(books.values());
338     if (ch == 1) {
339         Collections.sort(list);
340     } else {
341         list.sort(Comparator.comparing(Book::getAuthor));
342     }
343     for (Book b : list) {
344         b.displayBookDetails();
345         System.out.println("-----");
346     }
347 }
348
349 public void menu() {
350     while (true) {

```

```

Rakesh_Assingment.ipynb  ResultManager.java 7  LibrarySystem.java 9+ X
JAVA > LibrarySystem.java > Language Support for Java(TM) by Red Hat > LibrarySystem > Book > getBookId()
5 public class LibrarySystem {
149     static class LibraryManager {
349         public void menu() {
351             System.out.println("\n===== City Library Digital Management System =====");
352             System.out.println("1. Add Book");
353             System.out.println("2. Add Member");
354             System.out.println("3. Issue Book");
355             System.out.println("4. Return Book");
356             System.out.println("5. Search Books");
357             System.out.println("6. Sort Books");
358             System.out.println("7. Exit");
359             System.out.print("Enter choice: ");
360
361             int ch = sc.nextInt();
362             switch (ch) {
363                 case 1 ->
364                     addBook();
365                 case 2 ->
366                     addMember();
367                 case 3 ->
368                     issueBook();
369                 case 4 ->
370                     returnBook();
371                 case 5 ->
372                     searchBooks();
373                 case 6 ->
374                     sortBooks();
375                 case 7 -> {
376                     saveToFile();
377                     System.out.println("Exiting program. Data saved.");
378                     return;
379                 }
380                 default ->
381                     System.out.println("Invalid choice!");
382             }
383         }
384     }
385 }
386
Run main | Debug main | Run | Debug
387 public static void main(String[] args) {
388     LibraryManager manager = new LibraryManager();
389     manager.menu();
390 }
391 }
392

```

# Output:

```
PROBLEMS 21 OUTPUT DEBUG CONSOLE TERMINAL PORTS POSTMAN CONSOLE SPELL CHECKER 11

===== City Library Digital Management System =====
1. Add Book
2. Add Member
3. Issue Book
4. Return Book
5. Search Books
6. Sort Books
7. Exit
Enter choice: 1
Enter Book ID: 01
Enter Title: Java
Enter Author: Rakesh
Enter Category: Coding
Book added successfully!

===== City Library Digital Management System =====
1. Add Book
2. Add Member
3. Issue Book
4. Return Book
5. Search Books
6. Sort Books
7. Exit
Enter choice: 2
Enter Member ID: 02
Enter Name: Tina
Enter Email: 64@gmail.com
Member added successfully!

===== City Library Digital Management System =====
1. Add Book
2. Add Member
3. Issue Book
4. Return Book
5. Search Books
6. Sort Books
7. Exit
Enter choice: 7
Exiting program. Data saved.

d:\RAKESH\VSC\JAVA>
```

# Explanation

## Objective:

To develop a simple digital library management system using **File Handling** and **Java Collections Framework (JCF)**.

It helps to **add, issue, return, search, and sort books**, while maintaining **member details**.

## Key Concepts Used:

### 1. Classes & Objects:

- Book → Stores book details (ID, title, author, category, status).
- Member → Stores member details (ID, name, email, issued books).
- LibraryManager → Performs operations like add, issue, return, search, and sort.

### 2. File Handling:

- Uses FileReader, FileWriter, BufferedReader, and BufferedWriter to **save and load data** from books.txt and members.txt.
- Ensures that data is **persistent** even after the program closes.

### 3. Java Collections Framework (JCF):

- HashMap → Stores books and members using their IDs as keys.
- ArrayList → Used for managing lists of issued books.

### 4. Interfaces:

- Comparable<Book> → Enables sorting books by **title**.
- Comparator → Used for sorting by **author**.

### 5. Exception Handling:

- Handles errors like **invalid input, file not found, or duplicate IDs** using try-catch blocks.