

Student Name : Omar Sebri

Student Number : 3722350

```
/**
Represents a Book in a personal library
@author Omar Sebri : 3722350
*/
public class Book{

    /**
    The ISBN of the book
    */
    private final long ISBN;

    /**
    The title of the book
    */
    private final String TITLE;

    /**
    The author of the book
    */
    private final String AUTHOR;

    /**
    The number of times the book has been read
    */
    private int read;

    /**
    Constructs a book with the specific ISBN, title and author.
    @param isbn The ISBN of the book.
    @param title The title of the book.
    @param author The author of the book.
    @param read The number of times the book has been read.
    */
    public Book(long isbn, String title, String author, int read){
        ISBN = isbn;
        TITLE = title;
        AUTHOR = author;
        this.read = read;
    }

    /**
    This method returns the ISBN of the book.
    @return The ISBN of the book.
    */
}
```

```

    */
    public long getIsbn(){
        return ISBN;
    }

    /**
     This method returns the title of the book.
     @return The title of the book.
     */
    public String getTitle(){
        return TITLE;
    }

    /**
     This method returns the author of the book.
     @return The author of the book.
     */
    public String getAuthor(){
        return AUTHOR;
    }

    /**
     This method returns the number of times the book has been read.
     @return The number of times the book has been read.
     */
    public int getRead(){
        return read;
    }

    /**
     This method changes the number of times the book has been read.
     @param readIn The number of times the book has been read.
     */
    public void setRead(int readIn){
        read = readIn;
    }
}

```

```

import java.util.Scanner;

/**
 Represents a personal library.
 @author Omar Sebri : 3722350
 */
public class Library{

    /**

```

```

A collection of books.
*/
private Book[] library;

/**
Constructs a library with the specified collection of books.
@param library A collection of books.
*/
public Library(Book[] library){
    this.library = library;
}

/**
Constructs a library by reading in the specified input.
@param scanIn Input containing a list of books.
*/
public Library(Scanner scanIn){
    library = new Book[scanIn.nextInt()];
    scanIn.nextLine();

    for(int i = 0; i < library.length; i++){
        Scanner scan = new Scanner(scanIn.nextLine());
        scan.useDelimiter(",");
        long isbn = scan.nextLong();
        String title = scan.next();
        String author = scan.next();
        int read = scan.nextInt();
        library[i] = new Book(isbn, title, author, read);
    }
}

/**
This method determines how many books occur in one library and
not the other
@param other Other library to compare books in this library.
@return The number of unique books.
*/
public int findUnique(Library other){
    //TO DO: Complete this method
    int unique=0;
    if(this.library.length==0)
        return other.library.length ;
    else if(other.library.length==0)
        return this.library.length ;
    else {
        for(int i=0; i<library.length-1;i++){
            if(library[i]!=library[i+1]){
                unique+=1;
            }
        }
    }
}

```

```

    }
}
return unique;
}
}

/**
This method merges two libraries (includes both if duplicates).
@param other Library to merge with this library.
@return The library containing all contents of both libraries.
*/
public Library merge(Library other){
    //TO DO: Complete this method
    int i=0;
    int j=0; int k=0;
    int lib_length=this.library.length + other.library.length ;
    Book [] merged_lib = new Book[lib_length];
    while(i<this.library.length && j< other.library.length){
        if(this.library[i].getIsbn()<other.library[j].getIsbn()){
            merged_lib[k]=this.library[i];
            i++;
            k++;
        }
        else{
            merged_lib[k]=other.library[j];
            j++;
            k++;
        }
    }
    while(i<this.library.length){
        merged_lib[k]=this.library[i];
        i++;
        k++;
    }
    while(j< other.library.length){
        merged_lib[k]=other.library[j];
        j++;
        k++;
    }

    Library sorted_lib = new Library(merged_lib);
    return sorted_lib ;
}

/**
This method add a book in sorted order by ISBN to this library.
@param bookIn The book to be added to this library in sorted order.
*/

```

```

    public void addBook(Book bookIn){
        //TO DO: Complete this method
        Book [] temp = {bookIn};
        Library tempLib1 = new Library(temp);
        Library tempLib2 = merge(tempLib1);
        this.library = tempLib2.library ;
    }

    /**
    Creates a formatted print out of the library.
    @return Formatted print out of the library.
    */
    public String toString(){
        String result = "";
        for(int i = 0; i < library.length; i++){
            result += library[i].getIsbn() + "\t" +
                library[i].getTitle() + "\t" +
                library[i].getAuthor() + "\t" +
                library[i].getRead() + "\n";
        }
        return result;
    }
}
}

```

```

import java.util.Scanner;
import java.io.* ;
/**
Driver to test the Library class.
@author Omar Sebri
*/
public class TestLibrary{

    public static void main(String[] args)
    throws IOException
    {
        Scanner scs = new Scanner(System.in);
        System.out.print("enter the name of the file");
        String file_name = scs.nextLine();
        Scanner scan = new Scanner(new File(file_name));
        Library myLibrary = new Library(scan);
        Library yourLibrary = new Library(scan);
    }
}

```

```

Scanner sc = new Scanner(scan.nextLine());
sc.useDelimiter(",");
long isbn = sc.nextLong();
String title = sc.next();
String author = sc.next();
int read = sc.nextInt();
Book bookToAdd = new Book(isbn, title, author, read);
String result = file_name.substring(0, file_name.length()-4) +
"_output.txt" ;
/* Saves to a file under the name testfilename_output.txt */
File output = new File(result);
FileWriter fr = new FileWriter(output);

fr.write(myLibrary.toString());
fr.write(yourLibrary.toString());

fr.write("Number of unique books: " +
        myLibrary.findUnique(yourLibrary));

myLibrary.addBook(bookToAdd);
fr.write("My Library after adding a book: \n" +
        myLibrary);

fr.write("Merged Libraries: ");
fr.write(myLibrary.merge(yourLibrary).toString());
fr.close();
scs.close();

}
}

```

Test Case 1:

input(inp1.txt):

5

9780006716693,The Last Battle,Lewis,2

9780007591855,The Silmarillion,Tolkien,3

9780048232298,The Lord of the Rings,Tolkien,1

9780385491037,The Robber Bride,Atwood,3

9781400079179,The Da Vinci Code,Brown,2

3

9780048232298,The Lord of the Rings,Tolkien,1

9780062230621,The Confidence Code,Kay,0

9780385491037,The Robber Bride,Atwood,3

9780071079170,The Skin We're In,Cole,5

Output(inp1_output.txt):

9780006716693	The Last Battle	Lewis	2
9780007591855	The Silmarillion	Tolkien	3
9780048232298	The Lord of the Rings	Tolkien	1
9780385491037	The Robber Bride	Atwood	3
9781400079179	The Da Vinci Code	Brown	2
9780048232298	The Lord of the Rings	Tolkien	1
9780062230621	The Confidence Code	Kay	0
9780385491037	The Robber Bride	Atwood	3

Number of unique books: 4

My Library after adding a book:

9780006716693	The Last Battle	Lewis	2
9780007591855	The Silmarillion	Tolkien	3
9780048232298	The Lord of the Rings	Tolkien	1
9780071079170	The Skin We're In	Cole	5
9780385491037	The Robber Bride	Atwood	3
9781400079179	The Da Vinci Code	Brown	2

Merged Libraries:

9780006716693	The Last Battle	Lewis	2
9780007591855	The Silmarillion	Tolkien	3
9780048232298	The Lord of the Rings	Tolkien	1
9780048232298	The Lord of the Rings	Tolkien	1
9780062230621	The Confidence Code	Kay	0
9780071079170	The Skin We're In	Cole	5
9780385491037	The Robber Bride	Atwood	3
9780385491037	The Robber Bride	Atwood	3
9781400079179	The Da Vinci Code	Brown	2

Output Explanation:

We were able to print two libraries with some books in common, find out the number of unique books, add a book to them, merge the two libraries in sorted ascending order then print them

Test Case 2:

input(inp2.txt):

2

9780006716693,The Last Battle,Lewis,2

9780007591855,The Silmarillion,Tolkien,3

2

9780006716693,The Last Battle,Lewis,2

9780007591855,The Silmarillion,Tolkien,3

9780385491037,The Robber Bride,Atwood,3

Output(inp2_output.txt):

9780006716693 The Last Battle Lewis 2

9780007591855 The Silmarillion Tolkien 3

9780006716693 The Last Battle Lewis 2

9780007591855 The Silmarillion Tolkien 3

Number of unique books: 1

My Library after adding a book:

9780006716693 The Last Battle Lewis 2

9780007591855 The Silmarillion Tolkien 3

9780385491037 The Robber Bride Atwood 3

Merged Libraries:

9780006716693 The Last Battle Lewis 2

9780006716693 The Last Battle Lewis 2

9780007591855 The Silmarillion Tolkien 3

9780007591855 The Silmarillion Tolkien 3

9780385491037 The Robber Bride Atwood 3

Output Explanation

We tried Similar Libraries and we ended up with the number of unique books being 1 and the book added successfully and the two libraries merged successfully in ascending order

Test Case 3:
input(inp3.txt):

```
1
9780006716693,The Last Battle,Lewis,2
1
9780006716693,The Last Battle,Lewis,2
9780006716693,The Last Battle,Lewis,2
```

output(inp_output3.txt):

```
9780006716693      The Last Battle  Lewis   2
```

```
9780006716693      The Last Battle  Lewis   2
```

Number of unique books: 0

My Library after adding a book:

```
9780006716693      The Last Battle  Lewis   2
```

```
9780006716693      The Last Battle  Lewis   2
```

Merged Libraries:

```
9780006716693      The Last Battle  Lewis   2
```

```
9780006716693      The Last Battle  Lewis   2
```

```
9780006716693      The Last Battle  Lewis   2
```

Test case explanation:

We tried two libraries that consist each of one book that is the same we ended with number of unique books as zero, book was added successfully and the libraries were merged successfully

Test Case 4:
input(inp4.txt):

```
1
9780006716693,The Last Battle,Lewis,2
0
```

9780006716693,The Last Battle,Lewis,2

Test Case 4:

output(inp_output4.txt):

9780006716693 The Last Battle Lewis 2

Number of unique books: 1

My Library after adding a book:

9780006716693 The Last Battle Lewis 2

9780006716693 The Last Battle Lewis 2

Merged Libraries:

9780006716693 The Last Battle Lewis 2

9780006716693 The Last Battle Lewis 2

Test case explanation:

We tried merging two libraries and one of them was empty and the other contained one book, we ended up with the number of unique books being 1 and the book was added successfully and we ended up with our own library when we tried merging

Test Case 5:

input(inp5.txt):

0

0

9780385491037,The Robber Bride,Atwood,3

output(inp5_output.txt):

Number of unique books: 0

My Library after adding a book:

9780385491037 The Robber Bride Atwood3

Merged Libraries:

9780385491037	The Robber Bride	Atwood3
---------------	------------------	---------

Test case explanation:

We had two empty libraries and a book to add as our input, we ended up not printing anything concerning the empty libraries, the number of unique books was 0 and finally we added a book and then the merged list contained that book only