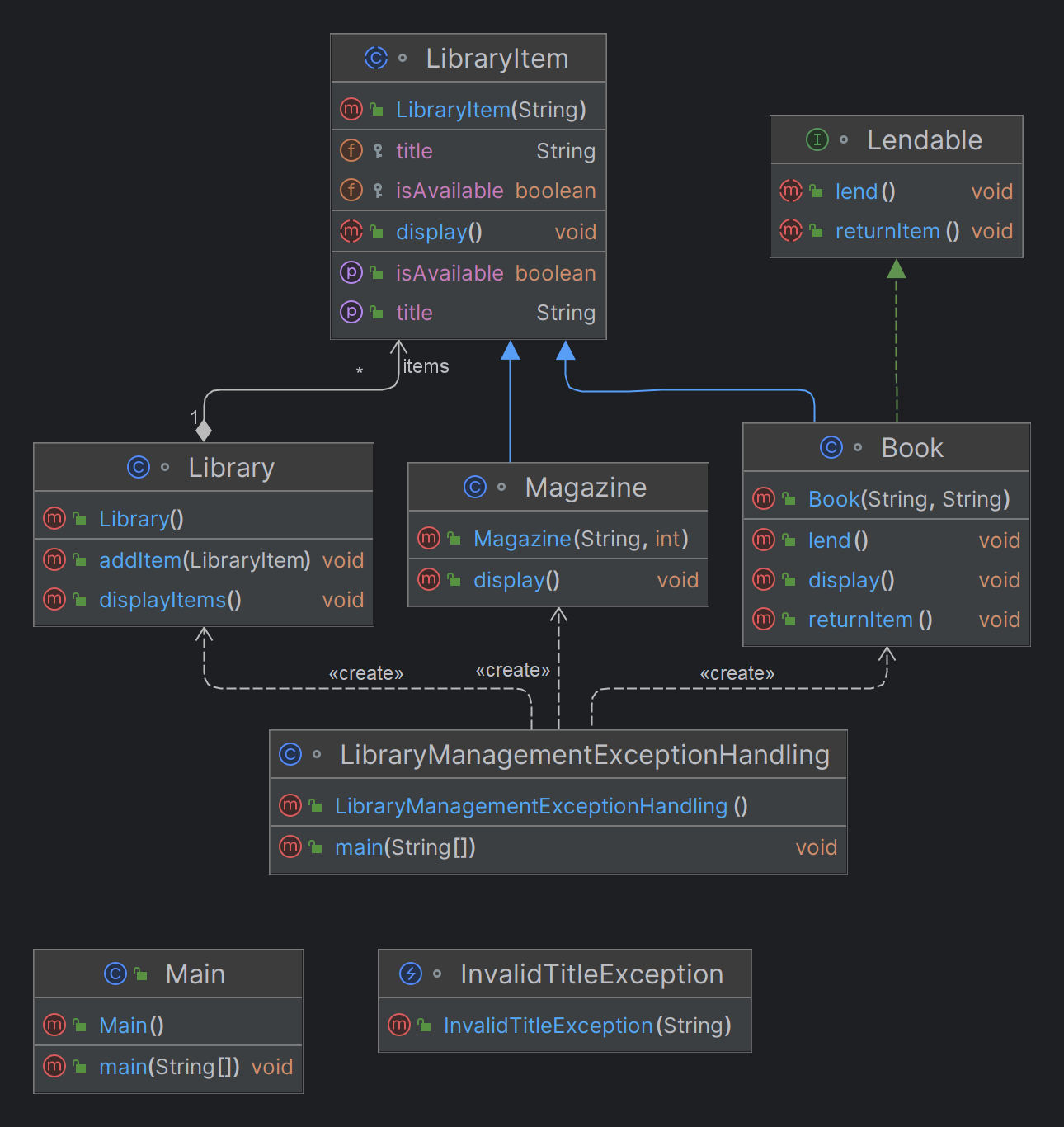
**Rupa Das**

**ID : 221-15-4952**

**Sec : 61-V**

****

import java.io.\*;  
import java.util.ArrayList;  
  
class Library {  
 ArrayList<LibraryItem> items;  
  
 public Library() {  
 this.items = new ArrayList<>();  
 }  
  
 public void addItem(LibraryItem item) {  
 items.add(item);  
 }  
  
 public void displayItems() {  
 System.*out*.println("Library Items:");  
  
 for (LibraryItem item : items) {  
 item.display();  
 System.*out*.println();  
 }  
 }  
}  
  
  
abstract class LibraryItem {  
 protected String title;  
 protected boolean isAvailable;  
  
 public LibraryItem(String title) {  
 this.title = title;  
 this.isAvailable = true;  
 }  
  
 public abstract void display();  
  
 public String getTitle() {  
 return title;  
 }  
  
 public boolean isAvailable() {  
 return isAvailable;  
 }  
  
 public void setAvailable(boolean available) {  
 isAvailable = available;  
 }  
}  
  
  
  
  
class Book extends LibraryItem implements Lendable {  
 private String author;  
  
 public Book(String title, String author) {  
 super(title);  
 this.author = author;  
 }  
  
 @Override  
 public void display() {  
 System.*out*.println("Book Title: " + title);  
 System.*out*.println("Author: " + author);  
 System.*out*.println("Availability: " + (isAvailable ? "Available" : "Not Available"));  
 }  
  
 @Override  
 public void lend() {  
 if (isAvailable) {  
 isAvailable = false;  
 System.*out*.println("Book '" + title + "' has been lent.");  
 } else {  
 System.*out*.println("Book '" + title + "' is not available for lending.");  
 }  
 }  
  
 @Override  
 public void returnItem() {  
 isAvailable = true;  
 System.*out*.println("Book '" + title + "' has been returned.");  
 }  
}  
  
  
  
  
class Magazine extends LibraryItem {  
 private int issueNumber;  
  
 public Magazine(String title, int issueNumber) {  
 super(title);  
 this.issueNumber = issueNumber;  
 }  
  
 @Override  
 public void display() {  
 System.*out*.println("Magazine Title: " + title);  
 System.*out*.println("Issue Number: " + issueNumber);  
 System.*out*.println("Availability: " + (isAvailable ? "Available" : "Not Available"));  
 }  
}  
interface Lendable {  
 void lend();  
 void returnItem();  
}  
  
  
  
class InvalidTitleException extends Exception {  
 public InvalidTitleException(String message) {  
 super(message);  
 }  
}  
  
  
class LibraryManagementExceptionHandling {  
 public static void main(String[] args) throws IOException {  
 Library library = new Library();  
  
 Book book1 = new Book("Java Programming", "John Doe");  
 library.addItem(book1);  
  
 Magazine magazine1 = new Magazine("Tech Today", 10);  
 library.addItem(magazine1);  
  
 library.displayItems();  
  
 // Writing library items to a text file  
 try {  
 FileWriter fileWriter = new FileWriter("library.txt");  
 BufferedWriter bufferedWriter = new BufferedWriter(fileWriter);  
  
 for (LibraryItem item : library.items) {  
 bufferedWriter.write(item.getClass().getSimpleName() + "," + item.getTitle());  
 bufferedWriter.newLine();  
 }  
  
 bufferedWriter.close();  
 System.*out*.println("Library items have been written to library.txt file.");  
 } catch (IOException e) {  
 System.*out*.println("Error writing to file: " + e.getMessage());  
 }  
  
 // Reading library items from a text file  
 try {  
 FileReader fileReader = new FileReader("library.txt");  
 BufferedReader bufferedReader = new BufferedReader(fileReader);  
  
 String line;  
 while ((line = bufferedReader.readLine()) != null) {  
 String[] data = line.split(",");  
 String className = data[0];  
 String title = data[1];  
  
 if (className.equals("Book")) {  
 Book book = new Book(title, "Unknown Author");  
 library.addItem(book);  
 } else if (className.equals("Magazine")) {  
 Magazine magazine = new Magazine(title, 0);  
 library.addItem(magazine);  
 }  
 }  
  
 bufferedReader.close();  
 System.*out*.println("Library items have been read from library.txt file.");  
 } catch (IOException e) {  
 System.*out*.println("Error reading from file: " + e.getMessage());  
 }  
  
 library.displayItems();  
 }  
}

**Output :**

Library Items:

Book Title: Java Programming

Author: John Doe

Availability: Available

Magazine Title: Tech Today

Issue Number: 10

Availability: Available

Library items have been written to library.txt file.

Library items have been read from library.txt file.

Library Items:

Book Title: Java Programming

Author: John Doe

Availability: Available

Magazine Title: Tech Today

Issue Number: 10

Availability: Available

Book Title: Java Programming

Author: Unknown Author

Availability: Available

Magazine Title: Tech Today

Issue Number: 0

Availability: Available