

```

1 import java.util.UUID;
2
3 public class LibraryItem {
4     protected String title;
5     protected String itemId;
6     protected boolean isCheckedOut;
7
8     public LibraryItem(String title) {
9         this.title = title;
10        // Generate a unique ID for each library item
11        this.itemId = UUID.randomUUID().toString().substring(beginIndex:0, endIndex:8);
12        this.isCheckedOut = false;
13    }
14
15    public void checkOut() {
16        this.isCheckedOut = true;
17        System.out.println(title + " has been checked out.");
18    }
19
20    public void returnItem() {
21        this.isCheckedOut = false;
22        System.out.println(title + " has been returned.");
23    }
24
25    public void displayDetails() {
26        System.out.println("Title: " + title);
27        System.out.println("Item ID: " + itemId);
28        System.out.println("Status: " + (isCheckedOut ? "Checked Out" : "Available"));
29    }
30 }

```

```

1 public class Book extends LibraryItem {
2     private String author;
3     private String isbn;
4
5     public Book(String title, String author, String isbn) {
6         super(title); // Call the constructor of LibraryItem
7         this.author = author;
8         this.isbn = isbn;
9     }
10
11    // Override to add book-specific details
12    @Override
13    public void displayDetails() {
14        System.out.println(x:"--- Book Details ---");
15        super.displayDetails(); // Call parent method
16        System.out.println("Author: " + author);
17        System.out.println("ISBN: " + isbn);
18    }
19 }

```

```

1 public class Magazine extends LibraryItem {
2     private int issueNumber;
3     private String publicationDate;
4
5     public Magazine(String title, int issueNumber, String publicationDate) {
6         super(title);
7         this.issueNumber = issueNumber;
8         this.publicationDate = publicationDate;
9     }
10
11    @Override
12    public void displayDetails() {
13        System.out.println(x:"--- Magazine Details ---");
14        super.displayDetails();
15        System.out.println("Issue Number: " + issueNumber);
16        System.out.println("Publication Date: " + publicationDate);
17    }
18 }

```

```

1 public class DVD extends LibraryItem {
2     private String director;
3     private int runtimeMinutes;
4
5     public DVD(String title, String director, int runtimeMinutes) {
6         super(title);
7         this.director = director;
8         this.runtimeMinutes = runtimeMinutes;
9     }
10
11     @Override
12     public void displayDetails() {
13         System.out.println(x:"--- DVD Details ---");
14         super.displayDetails();
15         System.out.println("Director: " + director);
16         System.out.println("Runtime: " + runtimeMinutes + " minutes");
17     }
18 }

```

```

1 public class ProjectTester {
2     Run | Debug
3     public static void main(String[] args) {
4         System.out.println(x:"=== Library Management System Test ===\n");
5
6         // Create a Book instance
7         Book aGameOfThrones = new Book("A Game of Thrones", "George R.R. Martin", "978-0553103540");
8         aGameOfThrones.displayDetails();
9         System.out.println();
10
11         // Create a Magazine instance
12         Magazine natGeo = new Magazine("National Geographic", 245, "September 2025");
13         natGeo.displayDetails();
14         System.out.println();
15
16         // Create a DVD instance
17         DVD interstellar = new DVD("Interstellar", "Christopher Nolan", 169);
18         interstellar.displayDetails();
19         System.out.println(x:"\n...Simulating checkout and return...\n");
20
21         // Test the methods
22         interstellar.checkOut();
23         System.out.println();
24         interstellar.displayDetails();
25         System.out.println();
26         interstellar.returnItem();
27         System.out.println();
28         interstellar.displayDetails();
29     }
30 }

```