```
package library;


public interface Issueable {
        boolean issuebook();
        String returnbooks();
   boolean isAvailable();
}
package library;


public class Book implements Issueable {
 private String title;
 private String author;
 private String ISBN;
 private boolean available;


 public Book(String title,String author,String ISBN) {
         this.title=title;
         this.author=author;
         this.ISBN=ISBN;
         this.available=true;
         }
@Override
public boolean issuebook() {
        if(!available) {
                return false;
        }
        available=false;
        return true;
}
@Override
public String returnbooks() {
        if(!available) {
                available=true;
                return "Return Successful";
        }
        return "Return failed:Book not borrowed";
}
@Override
public boolean isAvailable() {
        return available;
}
public String getTitle()
{
        return title;
}
public String getAuthor()
{
        return author;
}
public String getISBN()
{
        return ISBN;
}
}
package library;
import java.util.*;

public class Library {
private List<Book> bookList = new ArrayList<>();
public void includeBook(Book newBook) {
        bookList.add(newBook);
}
```

```
public String borrowBook(String ISBN) {
              for(Book b : bookList) {
                            if (ISBN.equals(b.getISBN())) {
            if (b.issuebook()) {
                return "Publication checkout successful";
            } else {
                return "Publication currently unavailable";
            }
        }
    }
    return "Publication not in collection";
}

public String bringBackBook(String ISBN) {
    for (Book b : bookList) {
        if (ISBN.equals(b.getISBN())) {
            return b.returnbooks();
        }
    }
    return "Publication not in collection";
}

public int collectionSize() {
    return bookList.size();
}

public List<Book> getAllPublications() {
    return bookList;
}
}
```

```
package library;

import static org.junit.jupiter.api.Assertions.*;
import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.Test;

public class LibraryUnitTests {

    private Library libraryInstance;
    private Book samplePublication;

    @BeforeEach
    void setup() {
        libraryInstance = new Library();
        samplePublication = new Book("Test Book", "Test Author", "TEST-123");
        libraryInstance.includeBook(samplePublication);
    }

    @Test
    void testBookBorrowing() {
        String result = libraryInstance.borrowBook("TEST-123");
        assertEquals("Publication checkout successful", result, "Book should be borrowed successfully");
        assertFalse(samplePublication.isAvailable(), "Book should be unavailable after being borrowed");
    }

    @Test
    void testBookReturning() {
        libraryInstance.borrowBook("TEST-123");
        String returnResult = libraryInstance.bringBackBook("TEST-123");
        assertEquals("Return Successful", returnResult, "Book should be returned successfully");
        assertTrue(samplePublication.isAvailable(), "Book should be available after return");
    }

    @Test
    void testAvailabilityStatus() {
        assertTrue(samplePublication.isAvailable(), "Book should be available initially");
        libraryInstance.borrowBook("TEST-123");
        assertFalse(samplePublication.isAvailable(), "Book should be unavailable after borrowing");
    }
}
```