Test Plan Library Management System

Team Member 1 :-

NAME : YUGABHARATHI E

SRN : PES1UG22CS716

Team Member 2 :-

NAME : VISHAL KUMAR RAHUL

SRN : PES1UG22CS700

1. Introduction

The Library Management System is designed to automate and streamline library operations such as student and book management, book issuing, and returning processes. It aims to enhance the efficiency of library operations while providing a user-friendly interface for librarians and students.

1.1 Scope In

Scope: -

User

Manageme

nt:

Registration

and

manageme

nt of

student

information

.

- Book Management:

Addition, deletion, and management of books, including checking stock status.

- **Book Issuing and Returning:**Streamlined process for issuing and returning books to and from students.
- **Data Retrieval:**Display of students, books, and issued books information.
- **Validation:** Input validation to prevent incorrect data entry and handle errors gracefully.

Out of Scope:

- Integration with external library networks.
- Advanced analytics for book lending patterns.
- Mobile app integration.

1.2 Quality Objectives

The goal is to develop a secure, efficient, and easy-touse Library Management System that ensures minimal errors and optimal performance. The system must handle data operations smoothly and display accurate information while preventing unauthorized data access.

1.3 Roles and Responsibilities

- **Project Manager:** Oversees the project timeline, allocates resources, and ensures timely deliverables.
- **Lead Developer:** Manages the overall system architecture, code quality, and feature development.
- **Backend Developer:**

Handles database integration and operations, including data validation and transaction management.

- **Frontend Developer:**

Designs and implements the graphical user interface using Tkinter.

- Product Tester:

Performs unit, integration, and system testing, documenting results and reporting issues.

---

2. Test Methodology

2.1 Test Levels -

Unit Testing:

Testing individual functions such as student registration, book addition, book issuing, and validation mechanisms.

- Integration Testing:

Ensuring components like book issuing, student data management, and data validation mechanisms work together seamlessly.

- System Testing:

Conducting end-to-end testing of the Library Management System to ensure all features function correctly in a real-world scenario.

- User Acceptance Testing (UAT):

Confirming the system meets the needs and expectations of users such as librarians and students.

---

3. Test Deliverables -

Test Plan:

A document outlining the testing process, strategies, and methodologies.

- Test Cases:

Detailed test scenarios covering all functionalities, including user and book management.

- Bug Reports:

Comprehensive documentation of identified bugs, their severity, and current status.

- Test Scripts:

Scripts for automated testing (if applicable) to cover key functions like data entry and retrieval.

- Test Summary Report:

A summary detailing test outcomes, issues found, and overall system performance.

---

4. Resource & Environment Needs

4.1 Testing Tools

- **MySQL Workbench**: Used for managing and validating the database structure and operations, ensuring the accuracy of data storage and retrieval.
- **PyCharm/VS Code**: An integrated development environment (IDE) for writing, debugging, and testing the Python code efficiently.
- **Tkinter**: The Python library used for creating the graphical user interface (GUI) of the Library Management System. It's tested directly within the Python environment.

- **Python Unit Test Library**: For writing and running unit tests to validate individual components of the system, such as data insertion and retrieval.
- **Manual Testing**: Conducted using the GUI interface to check all functionalities like adding and deleting students/books, issuing and returning books.

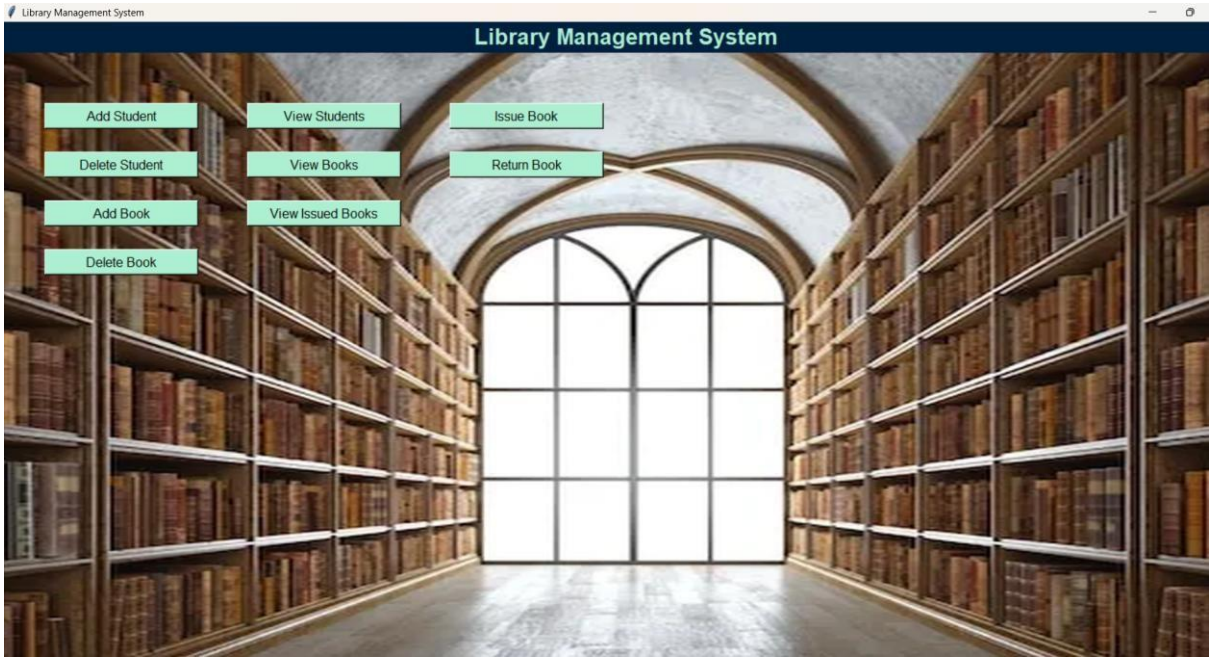
4.2 Test Environment

- **Operating System**: Windows 10 or later / macOS / Linux
- **Python Version**: Python 3.10 or later, configured with necessary libraries (mysql.connector, tkinter, etc.)
- **Database**: MySQL 8.0, running locally or on a test server to handle library data operations.
- **Hardware Requirements**:
    - Processor: Intel Core i3 or equivalent
    - RAM: 4GB minimum (8GB recommended)
    - Disk Space: At least 500MB free space for code and database

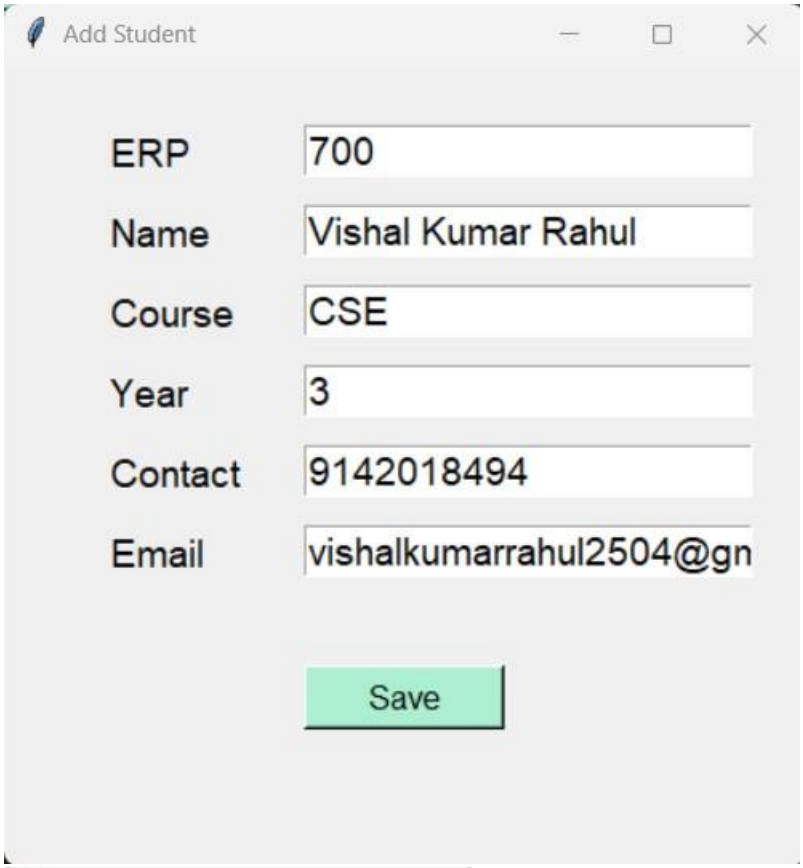
- **Network Requirements**: Local network access for connecting to the MySQL database if it's hosted separately.

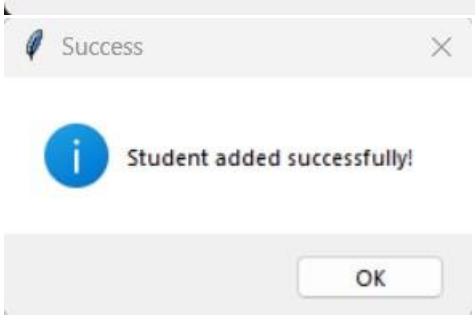
TRIGGERS	Update book Quantity and restore book quantity
JOINS	Issued books to student id and book id

RESULTS:-



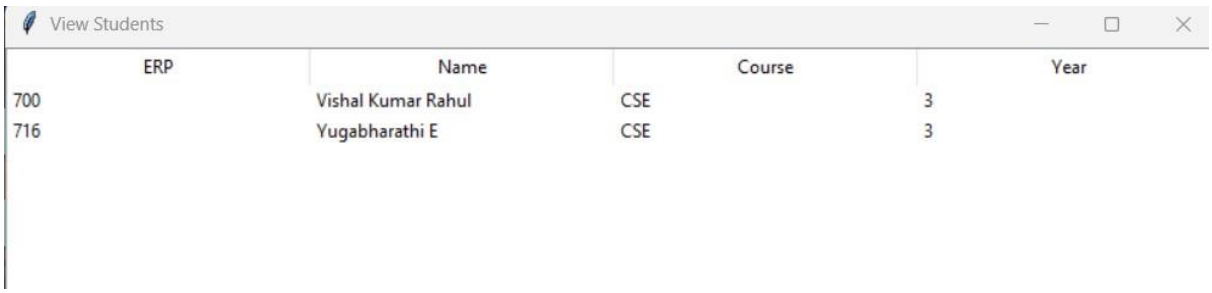
ADD STUDENT:-



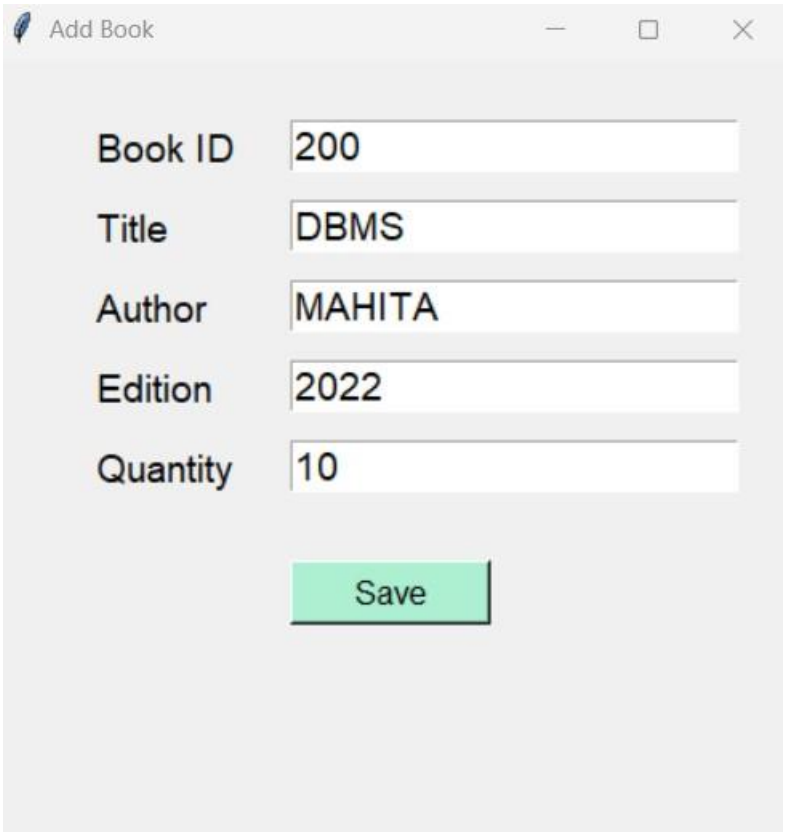


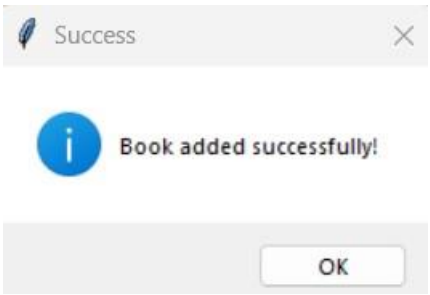

```
mysql> select * from students;
+-----+--------------------+--------+------+------------+--------------------------------+--------+
| ERP | Name               | Course | Year | Contact    | Email                          | NoBook |
+-----+--------------------+--------+------+------------+--------------------------------+--------+
| 700 | Vishal Kumar Rahul | CSE    |    3 | 9142018494 | vishalkumarrahul2504@gmail.com |      0 |
| 716 | Yugabharathi E     | CSE    |    3 | 8296226222 | eyuga@gmail.com                |      0 |
+-----+--------------------+--------+------+------------+--------------------------------+--------+
2 rows in set (0.00 sec)
```

VIEW STUDENTS:



ADD BOOK:

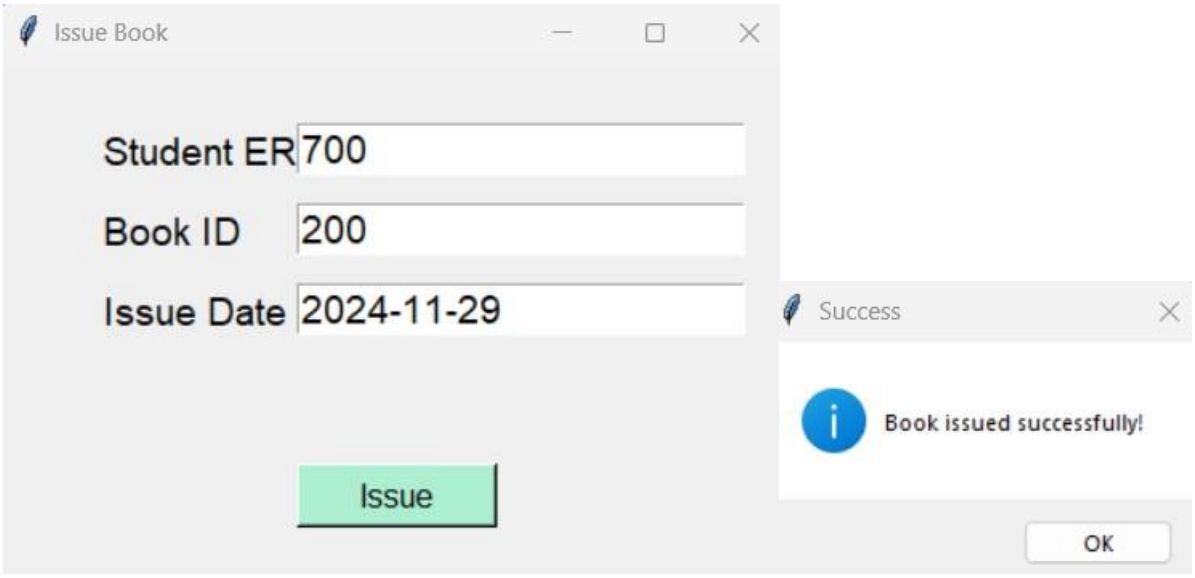





```
mysql> select * from books;
+--------+-------+--------+---------+----------+-----------+
| BookID | Title | Author | Edition | Quantity | Status    |
+--------+-------+--------+---------+----------+-----------+
|    200 | DBMS  | MAHITA | 2022    |       10 | Available |
+--------+-------+--------+---------+----------+-----------+
1 row in set (0.00 sec)

mysql>
```

View books:

ISSUE BOOKS:



SHOW ISSUE BOOKS:-

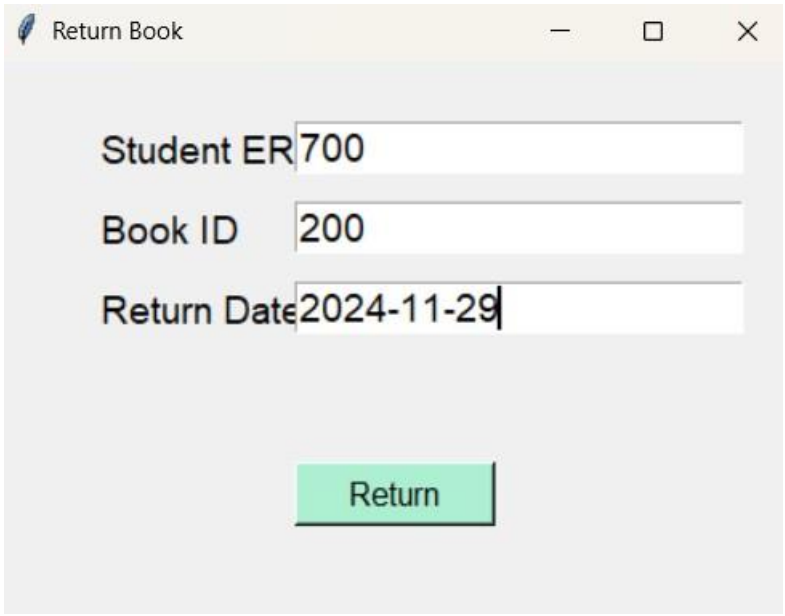

```
mysql> select * from issuedbooks;
+------+--------+------------+------------+
| ERP  | BookID | IssueDate  | ReturnDate |
+------+--------+------------+------------+
|  700 |    200 | 2024-11-29 | NULL       |
+------+--------+------------+------------+
1 row in set (0.00 sec)

mysql>
```

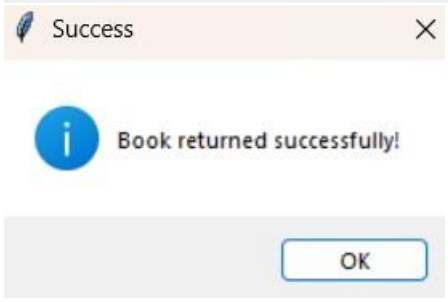
```
mysql> select * from students;
+-----+--------------------+--------+------+------------+--------------------------------+--------+
| ERP | Name               | Course | Year | Contact    | Email                          | NoBook |
+-----+--------------------+--------+------+------------+--------------------------------+--------+
| 700 | Vishal Kumar Rahul | CSE    |    3 | 9142018494 | vishalkumarrahul2504@gmail.com |      1 |
| 716 | Yugabharathi E     | CSE    |    3 | 8296226222 | eyuga@gmail.com                |      0 |
+-----+--------------------+--------+------+------------+--------------------------------+--------+
2 rows in set (0.00 sec)
```

TRIGGERS:-

Here trigger is used to update and restore books.

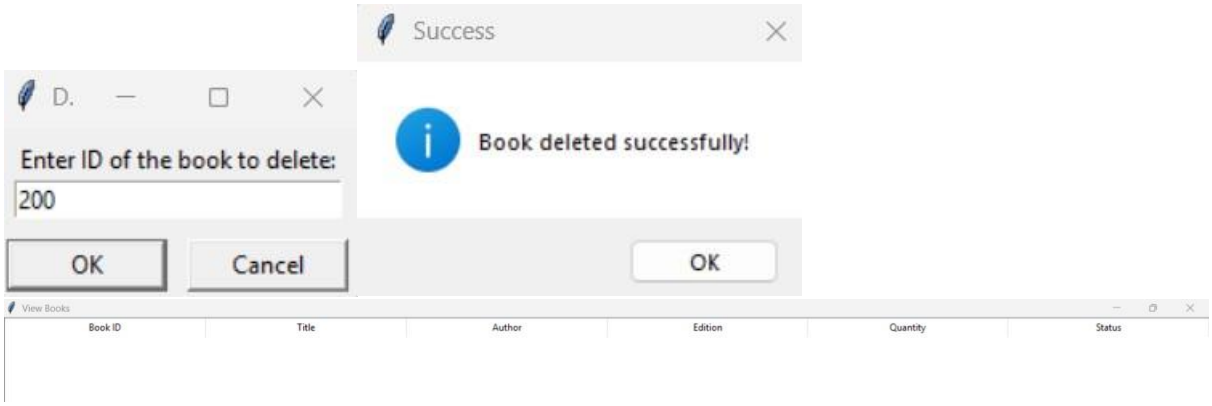
RETURN BOOKS:-





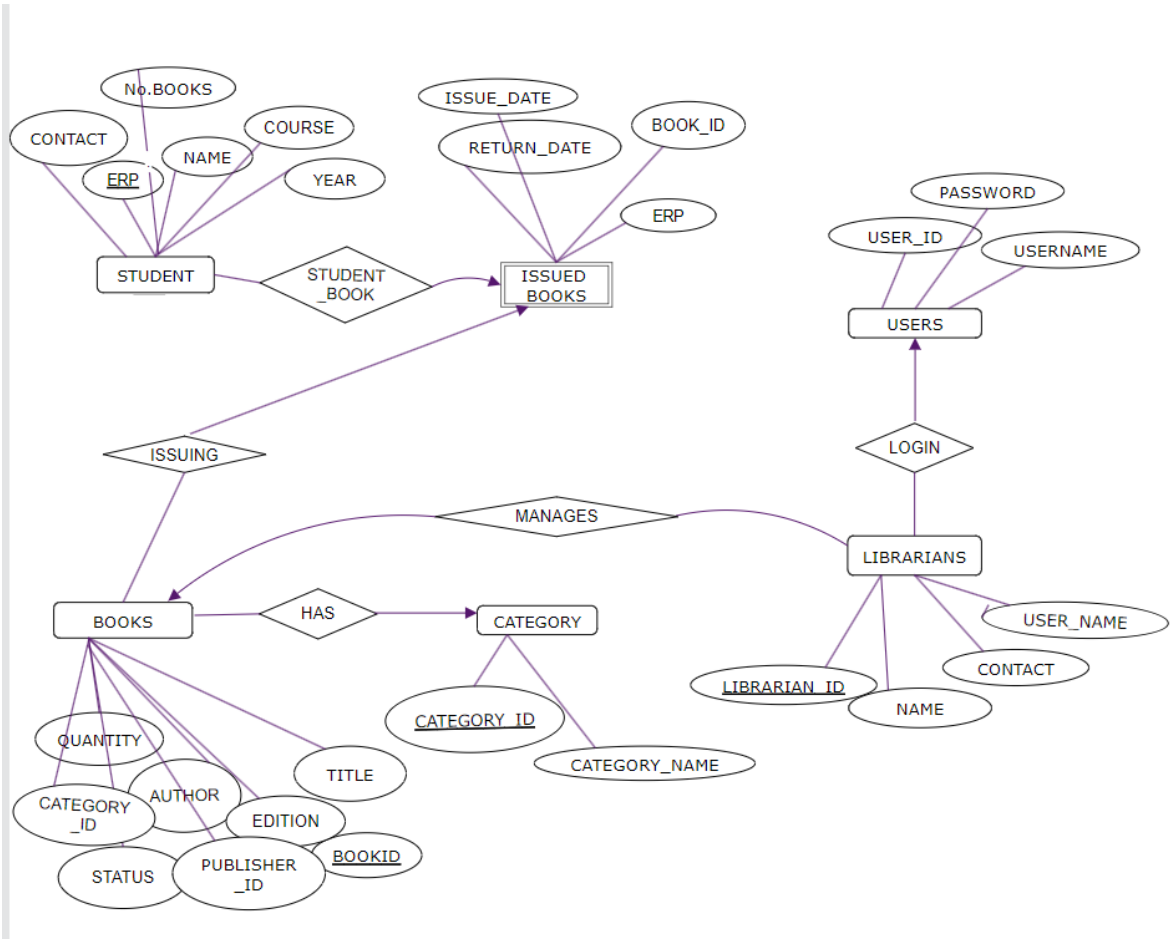
Here trigger is used which restore the book quantity from 9 to 10 after returning book.

DELETE BOOKS:




```
mysql> select * from books;
Empty set (0.00 sec)

mysql>
```

ER DIAGRAM:


No.BOOKS
CONTACT
COURSE
NAME
ERP
YEAR
STUDENT
STUDENT_BOOK
ISSUE_DATE
RETURN_DATE
BOOK_ID
ERP
ISSUED BOOKS
PASSWORD
USER_ID
USERNAME
USERS
ISSUING
LOGIN
MANAGES
LIBRARIANS
BOOKS
HAS
CATEGORY
USER_NAME
CONTACT
LIBRARIAN_ID
NAME
CATEGORY_ID
CATEGORY_NAME
QUANTITY
TITLE
AUTHOR
CATEGORY_ID
EDITION
BOOKID
STATUS
PUBLISHER_ID