****

**"Library Database System"（II）**

**Based on Access Database and SQL**

**Course Code：ISOM3000(003)**

**Course Title：Database Management**

**Date of Submission：29 November，2023**

**Group Members：**

**BC209436,YANG XI**

**BC103272 SOU SAI IENG**

**BC207013 IEONG KAI IN**

**BC105601 ZHENG TONGYU**

**BC207981 CHAN IAMFAI**

**BC204598 Deng JunTao**

This "Library Database System" is a simple system designed and developed with the purpose and background of systematizing the work of library circulation.

The system mainly realizes the registration, preservation, statistics and query functions for the basic information of books and readers, in addition to the transactions information is registered, preserved, statistics and can query.

## General system analysis and design

According to the library circulation scenes for the convenience of librarians, the system can be divided into Data Management, Data Manipulation, Data Inquiry three major boards.

1. Data Management：It mainly contains the records of books and readers' basic information, and you can continuously add new books or readers' information to the center.
2. Data Manipulation：It mainly contains the operation and management of the two processes of borrowing and returning books, which is the most core part of the whole system.
3. Data Inquiry：Including book information, reader information and borrowing overdue query, we can use keywords to book information, reader information query, but also can query all the borrowing overdue records.

**Book Management**

**Data Management**

**Reader Management**

**Library**

**Database**

**System**

**Borrow**

**Data Management**

**Return**

**Book Inquiry**

**Data Management**

**Reader Inquiry**

## Analysis and Design of Database

**Over List**

1. Database Logical Structure Design
   1. BOOK ENTITY

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BOOK ENTITY** | | | | |
| **Attributes** | **Data Type** | **Length** | **Required** | **Indexed** |
| ISBN | Short Text | 255 | Yes | No |
| Book ID | Short Text | 255 | Yes | No |
| Book Name | Short Text | 255 | Yes | No |
| Publisher ID | Short Text | 255 | Yes | No |
| Author 1 | Short Text | 30 | No | No |
| Author 2 | Short Text | 30 | No | No |
| Author 3 | Short Text | 30 | No | No |
| Publisher Year | Short Text | 5 | Yes | No |
| Edition | Number | Long Integer | Yes | No |
| Status | Short Text | 255 | No | No |
| Due Date | Date/Time |  | Yes | No |

* 1. READER ENTITY

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **READER ENTITY** | | | | | |
| **Attributes** | **Data Type** | **Length** | **Required** | **Indexed** | |
| Reader ID | Short Text | 255 | Yes | Yes(No duplicates) |  |
| Reader Name | Short Text | 30 | Yes | No | |
| Address | Short Text | 30 | Yes | No | |
| Grender | Short Text | 2 | No | No | |
| Quota | Number | Long Integer | Yes | No | |
| No of Check out | Number | Long Integer | Yes | No | |
| Telephone Name | Short Text | 30 | Yes | No | |
| Email Address | Short Text | 30 | Yes | No | |

* 1. TRANSACTION ENTITY

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TRANSACTION ENTITY** | | | | |
| **Attributes** | **Data Type** | **Length** | **Required** | **Indexed** |
| Tran ID | Self-number | Long Integer |  | Yes(No duplicated) |
| Reader ID | Short Text | 255 | YES | Yes(Duplicated) |
| Book ID | Short Text | 255 | YES | Yes(Duplicated) |
| Employee ID | Short Text | 255 | YES | Yes(Duplicated) |
| Borrow Date | Date/Time |  | YES | No |
| Return Date | Date/Time |  | NO | No |

* 1. EMPLOYEE ENTITY

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EMPLOYEE ENTITY** | | | | |
| **Attributes** | **Data Type** | **Length** | **Required** | **Indexed** |
| Employee ID | Short Text | 255 | No | Yes(No duplicates) |
| Employee Name | Short Text | 255 | No | No |
| Position | Short Text | 255 | No | No |

* 1. PUBLISHER ENTITY

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PUBLISHER ENTITY** | | | | |
| **Attributes** | **Data Type** | **Length** | **Required** | **Indexed** |
| Publisher ID | Short Text | 255 | Yes | Yes(No duplicates) |
| Publisher Name | Short Text | 255 | Yes | No |
| Publisher Address | Short Text | 255 | Yes | No |
| City | Short Text | 255 | Yes | No |
| Country | Short Text | 255 | Yes | No |

### 图示 描述已自动生成

## Design of Queries

A large number of queries are used in this library management system. The design of this database consists mainly of select queries, append queries and update queries.

表格

低可信度描述已自动生成

1. Select queries：A large number of selective queries are used in this database, including Book conditions Searching, Book information searching, Borrow book information searching, Overdue Inquiry, List of Overdue Student, Reader conditions Searching, Reader information searching.

文本

描述已自动生成

1. Append queries： In the Data Manipulation section, the implementation of borrowing and returning books uses append queries to append operations to the transaction table.

手机屏幕截图

描述已自动生成

1. Updating queries：In the Data Manipulation section, the implementation of borrowing and returning books uses updating queries to append operations to the transaction table.

手机屏幕截图

描述已自动生成

## Design and Implementation of Data Management

In the Data Management part, this system is the system administrator’s function to handle book information or reader’s account information.

图形用户界面

描述已自动生成

**Part1 Book Management**

For the manager of the book, we first click “Book Management,” then the system will pop up the following page.

图形用户界面, 应用程序

描述已自动生成

In this page, we use the book "Is Your Summer Alright?" as an example. We input the basic information of the book, such as ISBN, Book ID, Book Name, Publisher ID, Author 1, Author 2, Author 3 (if any), Publisher year, edition, status, due date into the corresponding space. Click “Add” to add information about the book and click “delete” to delete it. When you want to enter information about this book, you can click “Add,” then click “Preserve” after entering the basic information of this book, then the book information will be entered into the system. And we can also use the interface to find a book, just enter the ISBN, Book ID, or Book name, then click “Book List” to view our book information. After clicking Book List, our page will go to the following page.

图形用户界面, 应用程序, 表格

描述已自动生成图形用户界面, 应用程序

描述已自动生成

You can see the information and status of all the books in the system in Book List. This all is the basic operation of Book Management.

**Part2 Reader Management**

For the manager of the reader, we first click “Reader Management,” then the system will pop up the following page.

图形用户界面, 应用程序, Word

描述已自动生成

In this page, we can enter the basic information of the reader to add the reader account, such as Reader ID, Reader Name, Address, Gender, Quota, No of Check Out, Telephone Number, Email Address. We can also use this interface to modify the reader information. When the reader's basic information is all entered, we click "Add" and then click "Preserve" to successfully add the reader's account. Of course, we can also use the interface to find readers, just enter the Reader ID or Reader Name, and then click “Reader List”, the page will go to the following page, and we will be able to see the reader information of this system.

图形用户界面, 应用程序, 表格

描述已自动生成

You can see the information of the reader, such as reader id, reader Name, address, gender, borrowing quota, no of check out, Telephone number and Email address. This is all the basic operation of Reader Management.

## Design and Implementation of Data Manipulation

In Data Manipulate part, the system records the process of reader’s borrowing and returning of the book.

图片包含 图形用户界面

描述已自动生成

**Part1 Borrow**

For the borrowing of the book, we firstly click the “borrow”, then the system will show a page like the following figure.

图形用户界面

描述已自动生成

Now we use the reader whose reader ID is bc20943 as an example, click the “search”, it will show the reader information. We use a select query here to link to the reader information.

图形用户界面, 应用程序, Word

描述已自动生成

图形用户界面

描述已自动生成

In order to record the book which will be borrowed, we click the “book”and use Book ID to search the information of the book that the reader want to borrow. We use a select query here to link to the Book information.

图形用户界面, 应用程序

描述已自动生成 图形用户界面

描述已自动生成

Next, click the “Borrow”, it will show the process in this figure.

图形用户界面, 应用程序, Word

描述已自动生成

Click “yes”and insert the “Employee ID” to record who finishes the transaction. Here we use an append query and an updating query to update the book status and update the transaction.

图片包含 表格

描述已自动生成图形用户界面, 文本, 应用程序

描述已自动生成

Updating query（Reader updated Borrow） append query（Borrow\_transaction）

图形用户界面

描述已自动生成

Finally, we can have the data related to this transaction in “Transaction” table.

表格

描述已自动生成

**Part2 Return**

First, click the “Return”

图片包含 图形用户界面

描述已自动生成

It will automatically show the date when the reader returns the book. Next, use the “Reader ID” and “Book ID” and click “Search” to find out the borrowed book information.

图形用户界面, 应用程序, 表格

描述已自动生成

Then click “Return”, it will show something like what in this figure. After that clicl “yes”, Here we use an append query and an updating query to update the book status and update the transaction.

表格

中度可信度描述已自动生成图形用户界面, 文本, 应用程序

描述已自动生成

Updating query（Reader updated Return） append query（Return\_transaction）

图形用户界面, 应用程序, Word

描述已自动生成

Finally, insert the “Employee ID” to record the transaction of returning.

图形用户界面

描述已自动生成

## Design and Implementation of Data Manipulation

In Data Inquiry part, this system provides the user with the function of checking the borrowing details. This system includes Book Inquiry, Reader Inquiry and Overdue List.

图形用户界面

描述已自动生成

**Part 1** Book Inquiry

For the Book Inquiry.When we click book inquiry, the following screen will be displayed. There are four inquiry methods. The user can choose to query the status of the book by entering the Book ID or Book Name,or author.

图形用户界面, 应用程序, 表格

描述已自动生成

The following is a query example (picture). After we enter the Book ID and click the searching button, we can check the borrowing status of the book. The same goes for other search methods. In this piece we use searching query.

图形用户界面, 文本, 应用程序, Word

描述已自动生成

1. Inquir by Book ID:Enter Book ID, then click Searching button.

图形用户界面, 应用程序, Word

描述已自动生成

1. Inquiry by Book Name: Enter Book Name, then click Searching button.

图形用户界面, 应用程序, 表格

描述已自动生成

1. Inquiry by Author1: Enter Author, then click Searching button.

图形用户界面, 应用程序

描述已自动生成

4. Users can also query through combinations, such as querying book authors and Book ID together.

图形用户界面, 应用程序, 表格

描述已自动生成

**Part 2** Reader Inquiry

For the Reader Inquiry par.Users can find the reader's borrowing status by entering Reader Name or Reader ID, or Address in this system. It is used exactly the same way as Book Iquiry，In this piece we use searching query.

图形用户界面, 文本, 应用程序, Word

描述已自动生成

图形用户界面, 应用程序

描述已自动生成

**Part 3** List of Overdue Student

We start by searching query to find all the overdue records and then list the overdue students and books.

图形用户界面

中度可信度描述已自动生成

图形用户界面, 文本

描述已自动生成

For the List of Overdue Student, users need to query by entering reader ID.

## SQL Statement

1. Creation of records for new students/books/publishers/employees
   1. **INSERT INTO `Reader`(`Reader ID`, `Reader Name`, `Address`, `Gender`, `Quota`, `No of Check Out`, `Telephone Number`, `Email Address`) VALUES ('ac20658',' YANG QIAN',' w23 3024',' F',' 5',' 2',' 68954726',' ac20658@umac.mo'), ('ac20832',' ZI YUXUAN',' w31',' F',' 4',' 0',' 63586452',' ac20832@umac.mo'), ('ac20965',' LIU LITONG',' w23 1059',' M',' 5',' 0',' 69852347',' ac20965@umac.mo'), ('bc20943',' YANG XI',' w23 3125',' F',' 3',' 5',' 63545729',' bc20943@umac.mo'), ('bc21058',' LIN YIXING',' w23 2078',' F',' 4',' 0',' 68900125',' bc21058@umac.mo');**
   2. **INSERT INTO `Book` (`ISBN`, `Book ID`, `Book Name`, `Publisher ID`, `Author 1`, `Author 2`, `Author 3`, `Publisher Year`, `Edition`, `Status`, `Due Date`) VALUES ('9787020115518', '01203', '你的夏天还好吗', '56898', '金爱灿', '薛舟', '', '1980', '2', 'out', '2023-12-24'), ('7532706907',' 02351',' 百年孤獨',' 45631',' 加西亚马尔克斯',' ',' ',' 1927',' 8',' in',' 2023-12-25'), (' 9787532751495',' 02458',' 羅生門',' 45631',' 芥川龙之介',' ',' ',' 2010',' 5',' out',' 2023-12-25'), ('9784101001548',' 15236',' 海辺のカフカ',' 12364',' 村上春树',' ',' ',' 2005',' 7',' out',' 2022-12-24'), ('7532129020',' 59564',' 兄弟',' 35694',' 余华',' ',' ',' 2005',' 3',' out',' 2022-12-25');**
   3. [**INSERT**](http://bida.fba.um.edu.mo/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/insert.html)**INTO `Publisher`(`Publisher ID`, `Publisher Name`, `Publisher Address`, `City`, `Country`)**[**VALUES**](http://bida.fba.um.edu.mo/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/miscellaneous-functions.html#function_values)**('12364 ','新潮社', '新潮社大楼',' 东京',' 日本'), ('35694 ','上海文藝出版社', '上海文藝出版社大楼',' 上海',' 中国'), ('45631 ','上海譯文出版社', '上海譯文出版社大楼',' 上海',' 中国'), ('54632 ','澳门文学出版社', '澳门文学出版社',' 澳门',' 中国'), ('56898 ','人民文学出版社', '人民文学出版社大楼',' 北京',' 中国');**
   4. **INSERT INTO `Employee`(`Employee ID`, `Employee Name`, `Position`) VALUES ('0001',' WU AILING',' chief administrative officer'), ('0002',' LI JIANHUA',' Administrative Assistant'), ('0003',' LI JING',' Administrative Assistant'), ('0004',' LIU JIAYU',' Administrative Assistant'), ('0005',' LI YUFEI',' Administrative Assistant');**
2. Perform check-out procedure for students when borrowing books
   1. For Transaction:**INSERT INTO [Transaction] ( [Reader ID], [Book ID], [Employee ID], [Borrow Date] ) SELECT [Forms]Borrow]![ReaderID] AS [Reader ID], [Forms]![Boorow\_book]![BookID] AS [Book ID], [Employee]![Employee ID] AS [Employee ID], Date() AS [Borrow Date];**
   2. For Reader、Book：

**UPDATE Reader, Book SET Reader.Quota = [Reader]![Quota]-1, Book.Status = "out", Book.[Due date] = DateAdd("m",1,Date())**

**WHERE (((Reader.[Reader ID])=[Forms]![Borrow]![ReaderID]) AND ((Book.[Book ID])=[Forms]![Boorow\_book]![BookID]));**

1. Use book ID to search the availability of a book(the results of the query must show the book name, name of author(s), name of publisher, location and status)

**SELECT Book.[Book Name], Book.[Author 1], Book.[Author 2], Book.[Author 3], Publisher.[Publisher Name] , Publisher.[Location],Book.Status**

**FROM Book，Publisher**

**JOIN Publisher ON Book.Publishe\_ID = Publisher.Publisher\_ID**

**WHERE Book\_ID = '02351';**

1. Use student no. to search the quota and the no. of books checked out(the results of the query must show student no., student name, quota and no. of books checked out) by a student

**SELECT Reader.Reader\_ID, Reader.Reader\_Name, Reader.Quota, Reader.No\_of\_Check\_Out**

**FROM Reader**

**WHERE Reader\_ID = 'bc209436';**

**5.** Find a reader who has the most books overdue. Note: you need to show the student no., student name and the no. of books overdue.

**SELECT t.Reader\_ID, t.Reader\_Name, COUNT(\*) AS Overdue\_Count**

**FROM Transaction AS t**

**JOIN Book AS b ON t.Book\_ID = b.Book\_ID**

**WHERE t.Return\_Date > b.Due\_Date**

**GROUP BY t.Reader\_ID, t.Reader\_Name**

**ORDER BY Overdue\_Count DESC**

**LIMIT 1;**

## Reports of Readers、Books、Overdue Readers

![图形用户界面

描述已自动生成

表格

描述已自动生成

图形用户界面

描述已自动生成

## Strengths of the System (Selling Points)

## 1. Clean and easy-to-understand interface with clear and distinct functionality.

## 2. The Data Management interface allows users to view, modify, update, and create new book/reader records.

## 3. The borrowing and returning page allows users to input Reader ID and Book ID to view detailed reader and book information.

## 4. The Data Inquiry feature permits the locking of desired books/readers based on single or multiple search criteria.

## 5. Real-time monitoring of overdue borrowers and generation of reports.

## Limitations and Future Development

## • Limitations: 1.Lack of a borrowing renewal feature; borrowing defaults to a one-time process.

## 2. Returning books currently requires users to input the borrowing date, which should be automated in the future.

## 3. After a book is recorded as borrowed, new borrowing records can still be created before the return, potentially leading to operational errors. Future optimizations and restrictions for querying borrowed books are needed.

## • Future Development:1. Implementation of a borrowing renewal feature, allowing users to extend the borrowing period.

## 2.Automation of the matching of return dates, enhancing user-friendliness.

## 3. Optimization of borrowing book queries, including restrictions on creating new borrowing records for books that haven't been returned.