Library Management System

Database Design Document

Table of Contents

1. Introduction

The Library Management System (LMS) is a system for librarians to search for the books to know about the availability of the corresponding book in a certain library branch.

The primary goal of LMS project is to provide a way for librarians to add new borrowers or check-in, check-out certain books depending on the criteria of the respective library. Fines can also be paid or updated in LMS. The design of database is based on information that is to be collected or has been collected in the past.

The data is stored in MySQL 5.5 on a HP Server running Windows 10. Access to the database is through a Java application interface which is run on NetBeans or Eclipse. The design is to allow the easy data entry access and querying to librarians.

The driving philosophy behind this system was to have an efficient, normalized database that would be allow easy data entry and access.

2. Assumptions

- All book id's are taken as 10-character ISBN numbers (i.e. some contain alpha characters) though 13-character sequence is also given in the data provided.
- All books that are loaned are considered to be returned on the specified ‘date_in’ provided in the table.
- Update Fines does not update the fines table if there is a book that is not yet returned.
- ‘Card_no’ is auto_incremented in the data base but while issuing, it is appended to ‘ID’ followed by zeroes and then the ‘card_no’.
- The digits part is 6-digit long. SSN is stored in the format ‘xxx-xx-xxxx’.

3. Data base Design

The data base is designed using MySQL 5.5 command line client. The tables are loaded based on the given data from csv files.

3.1 Schema Diagram

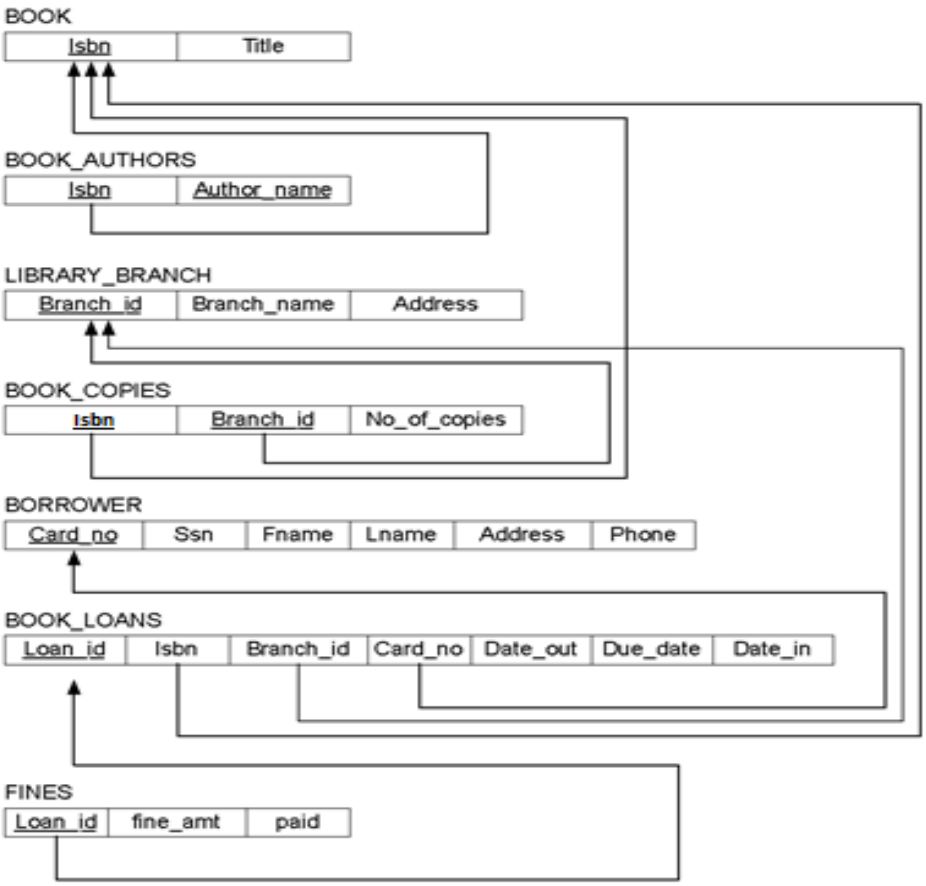

Figure 3.1: LMS Schema Diagram

This schema is used to implement the library database.

3.2 Book Table

Using books.csv file provided, ISBN (10-character) and Title columns are selected using the Load data command in mysql. Isbn is the primary key.

Table 1: BOOK

Column Name	Type
Isbn	Varchar
Title	Varchar

3.3 Authors Table

Using books.csv file provided, Author names are extracted using substrings in MySQL. Each Author is provided an author id which is auto_incremented. Author_id is primary key.

Table 2: AUTHORS

Column Name	Type
Author_id	Int
Fullname	Varchar
Title	Varchar
Fname	Varchar
Mname	Varchar
Lname	Varchar
Suffix	Varchar

3.4 Book Authors Table

Using books.csv file provided, ISBN is extracted. ISBN is a foreign key referenced to Isbn of Book Table. Author_id is a foreign key referenced to Author_id of Authors Table.

Table 3: BOOK_AUTHORS Table

Column Name	Type
Isbn	Varchar
Author_id	Int

3.5 Library Branch Table

Using library_branch.csv file provided, Branch_id, Branch_name, Address are extracted. Branch_id is the primary key for this table.

Table 4: LIBRARY_BRANCH Table

Column Name	Type
Branch_id	Int
Branch_name	Varchar
Address	Varchar

3.6 Borrower Table

Using borrower.csv file provided, the corresponding columns are extracted. Card_no is the primary key for this table. A person with an SSN can have only one card issued. Card_no is auto_incremented in the data base but while issuing it is appended to 'ID' followed by zeroes and then the card_no. The digits part is 6-digit long. SSN is stored in the format 'xxx-xx-xxxx'.

Table 5: BORROWER Table

Column Name	Type
Card_no	Int
Ssn	varchar
Fname	Varchar
Lname	Varchar
Address	Varchar
Phone	int

3.7 Book Copies Table

Using book_copies.csv file provided, the corresponding columns are extracted. Isbn is the foreign key which is referenced to Isbn of book table. Branch_id is the foreign key which is referenced to branch_id of library_branch table.

Table 6: BOOK_COPIES Table

Column Name	Type
Isbn	Varchar
Branch_id	Int
No_of_copies	int

3.8 Book Loans Table

Using book_loans.csv file provided, the corresponding columns are extracted. Isbn is the foreign key which is referenced to Isbn of book table. Branch_id is the foreign key which is referenced to branch_id of library_branch table. Card_no is the foreign key which is referenced to Card_no of borrower table. The default date_in is provided as '1885-01-01'. Loan_id is the primary key which is auto_incremented.

Table 7: BOOK_LOANS Table

Column Name	Type
Loan_id	Int
Isbn	Varchar
Branch_id	Int
Card_no	Int
Date_out	Date
Due_date	Date
Date_in	Date

3.9 Fines Table

Using book_loans table, the fines table is created for loan_id's which have date_in greater than due_date. The default paid value is 'false'. The fine amount is calculated using the below formula

Fine_amt = (the difference in days between the due_date and date_in) * $0.25

Loan_id is the foreign key which is referenced to loan_id of book_loans.

Fine amt is decimal with 2 precision.

Table 8: FINES Table

Column Name	Type
Loan_id	Int
Fine_amt	Decimal
Paid	boolean

4. GUI Design

GUI is designed using the window builder and java swings. NetBeans software is used.

The ‘design’ tab in the window builder application is used to design the GUI. The required labels, text fields, tables, buttons can be added to the application window. The corresponding code is generated in the source tab of the .java file.

The required functionality for each button is implemented in the code.

The project has two java files:

One is used for connecting to mysql database and getting the results for specified query.

The other file is the gui file.