# LIBRARY MANAGEMENT SYSTEM

# CS 631004

**Project Group No: 2**

**Project by:**


| | |
|---|---|
| **Francis D'cruz** | **fjd7@njit.edu** |
| **Joyce Chettiar** | **jpc68@njit.edu** |
| **Karthik Chandrashekar** | **kc69@njit.edu** |

## Description of implementation

We created a Menu Driven Library Management System, We used Python to build our GUI. To achieve the desired functions for our application, we used various SQL commands and operations. Operations like Joins, Functions, Conditional Statement were used to manipulate the data to achieve the desired output.

We designed a 3-page layout to allow easy access to the reader and the admin.

The first page is the **Login page**.
There are two users:

1. Reader
2. Admin

The Reader can log in using his/her **READERID.** Once the login is successful the reader will view the Reader Menu. This is our second page.

The Admin can log in using his/her **AdminID.** Once logged in They will view the Admin Menu and have admin rights. This is our Third page.

After login, the user will be taken to the respective**.**

The **Reader Menu** is the second page.

A reader can search for documents by ID or Publisher Name or title.

A reader can Borrow, Reserve, and or Return a Book using docid, BranchID, CopyNo, and ReaderID.

A Reader can also check the status of the books by using their ReaderID.

One can also search all the documents/books published by a certain author by using publisher's name.

The **Admin Menu** is our third page.

The Admin can add a document into the database using the document title, date published, and Publisher ID.

Admin can also check the top N number of books borrowed.

They can also calculate the fines for a given year for all the branch.

## Problem faced

- Once we started the implementation of the project we realized that our relational schema looked accurate theoretically but needed a little alteration while implementing the database structure.
- The referencing got a little confusing and it took us some time to understand it and implement it accurately.
- Insertion of the data took the most time as we had to make sure none of the constraints are violated.