INVESTIGATORY PROJECT

COMPUTER SCIENCE

By Kevin Immanuel

Sanchit Saini

Avinash Singh

LIBRARY MANAGEMENT SYSTEM

SYNOPSIS

1. PROJECT NAME: **LIBRARY MANAGEMENT SYSTEM**
2. GROUP PROJECT BY: **Kevin Immaneul, Avinash Singh and Sanchit Saini**
3. CLASS: **XII A**
4. SUBJECT: **COMPUTER SCIENCE (083)**
5. SCHOOL NAME: **KENDRIYA VIDYALAYA NO.2 JALLAHALLI EAST**
6. SUBMITTED TO: **MRS. LEENA (CS TEACHER)**
7. ACKNOWLEDGEMENT
8. PREFACE
9. TABLE OF CONTENTS
10. INTRODUCTION
11. CERTIFICATE
12. WORKING DESCRIPTION
13. HARDWARE AND SOFTWARE
14. HYPOTHESIS
15. CONTENT OF SYLLABUS USED
16. ANALYSIS
17. PROGRAM
18. OUTPUT
19. BIBLIOGRAPHY

ACKNOWLEDGEMENT

I would like to express my special thanks to my computer teacher **Mrs. Leena**, who give me the golden opportunity to do this wonderful project on the topic **Library Management System**. I am also very grateful to my other teachers and school teachers for helping me with the necessary equipment and valuable guidance.

Finally, I would like to thank my parents and close friends, who were always with me from imagination to the implementation of this project. Without their cooperation, this project would not have been completed. Only because of all of you, I was able to make my project successful and make it a good enjoyable experience.

Preface

> *" In order to be irreplaceable, one must always be different" – Coco Chanel*

This project is made by using two open-source programming languages, python and MySQL.

Python is a high-level, general-purpose programming language. Its design philosophy emphasises code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured, object-oriented and functional programming.

Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0. Python 3.0, released in 2008, was a major revision that is not completely backward-compatible with earlier versions. Python 2 was discontinued with version 2.7.18 in 2020.

Python consistently ranks as one of the most popular programming languages.

MySQL is an open-source relational database management system. Its name is a combination of "My", the name of co-founder Michael Widenius's daughter My, and "SQL", the acronym for Structured Query Language. MySQL is written in C and C++. MySQL works on many system platforms, including AIX, BSDi, FreeBSD, HP-UX, ArcaOS, eComStation, IBM i, IRIX, Linux, macOS, Microsoft Windows, etc.

TABLE OF CONTENTS

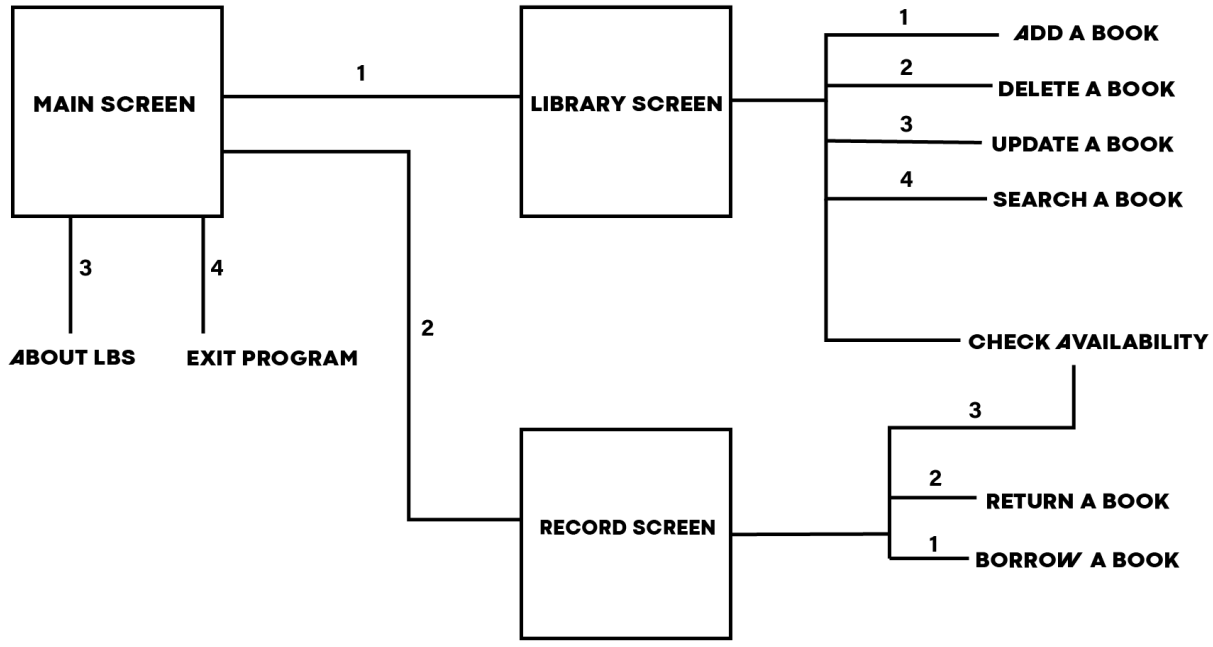
Introduction

This software helps to keep track of all the books in a school library and the records of students/teachers who borrow books from the library.

The user can borrow books in the library and save their details like their name, class, the book and the date. Every time a user wants to borrow/return a book it'll change the availability of the book in the library database.

The user can search for a book in the database by its name, author's name and genre.

This program will contain various more functions we will go through all of them in the documentation.



CERTIFICATE

THIS IS TO CERTIFY THAT **KEVIN IMMANUEL**, **AVINAH SINGH** AND **SANCHIT SAINI** OF **CLASS XII- 'A'** SUCCESSFULLY COMPLETED THEIR COMPUTER SCIENCE PROJECT ON **LIBRARY MANAGEMENT SYSTEM** UNDER THE ABLE GUIDANCE OF

MRS. LEENA P.G.T(COMPUTER SCIENCE)

SIGNATURE:

DATE :

Working Description

1. Creating the Database and required Tables

The program will be creating a database named `library` and it'll contain two tables. The first named `lib` will contain all the books in the library and their respective details.

The second table named `records` will contain all the records of the library.

> For example, 'borrowing/returning a book'.

The creation of the database and tables will be done in the following functions in the `createdb.py` file.

```
def createdb():
    cur.execute('create database library')
    cur.execute('use library')
    cur.execute('create table lib (snobook int primary key not null, name
varchar(30) not null, author varchar(30) not null, price int not null, pages int
not null, available boolean not null);')
    cur.execute('create table record (sno int primary key not null, dofentry date
not null, name varchar(30) not null, class varchar(10) not null, snobook int not
null')
    db.commit()
```

2. The Library functions

The `libraryfunc.py` file contains all the library functions below. These functions work with all the books in the library database.

2.1 ADD A BOOK

Inserting books in the `lib` table in the library database. The software will take **name**, **author**, **price** and **no. of pages** as the input and will insert the book in the table with an appropriate serial number (primary key).

The fucntion is named as `insertbook()` in the `librryfunc.py` file and is defined as...

```
def insertbook():
    while True:
        try:
            name = input('Enter the name of the book: ')
            author = input( 'Enter the name of the author: ')
            price = int(input('Enter the price of the book: '))
            pages = int(input('Enter the number of pages ini the book: '))
            break
        except ValueError:
            print('Value Error! Please try again!')
```

```
(lenofdata+1,name,author,price,pages,'1'))
    try:
        db.commit()
        print('Book added successfully!')
    except mysql.connector.Error as err:
            print('ERROR! please try again...')
            print(err)
            insertbook()
```

2.2 DELETE A BOOK

Removing a book from the library incase if the book isn't available in the library anymore. This function will just take the name of the book as the input.

The fucntion is named as `deletebook()` in the `libraryfunc.py` file and is defined as...

```
def deletebook():
    book = input('Enter the name of the book to delete: ')
    if checkbook(book) == True:
        for i in data:
            if i[1]==book:
                cur.execute('''delete from lib where name="%s"'''%(book))
                db.commit()
                print('Book deleted successfully!')
    else:
        print('Book not in database! Please try again!')
        deletebook()
```

2.3 UPDATE A BOOK

Change/update the details of a book. The function takes the name of the book as input.

The user can change any detail of the book like **name**, **author**, **price** and **genre**.

The function is named as `updatebook()` in the `libraryfunc.py` file and is defined as...

```
def updatebook():
    def inupbook(book,prop,newpropval):
        for i in data:
            if i[1]==book:
                cur.execute('''update lib set %s="%s" where name="%s"'''%
(prop,newpropval,book))
                db.commit()
    book = input('Enter the name of the book you would like to update: ')
    if checkbook(book) == True:
        print('''
        CHOOSE AN OPTION
        1) CHANGE NAME OF THE BOOK
        2) CHANGE AUTHOR OF THE BOOK
```

```
        3) CHANGE NUMBER OF PAGES
        4) CHANGE PRICE
        ''')
        optupbook = int(input('Enter your choice: '))
        if optupbook==1:
            newpropval = input('Enter new name of the book: ')
            inupbook(book,'name',newpropval)
        elif optupbook==2:
            newpropval = input('Enter new name of the author: ')
            inupbook(book,'author',newpropval)
        elif optupbook==3:
            newpropval = int(input('Enter new number of pages: '))
            inupbook(book,'pages',newpropval)
        elif optupbook==4:
            while True:
                try:
                    newpropval = int(input('Enter new price: '))
                    inupbook(book,'price',newpropval)
                except ValueError:
                    print('Please enter an integer')
                    updatebook()
        print('Book updated successfully!')
    else:
        print('Book not in database! Please try again')
        updatebook()
```

2.4. SEARCH A BOOK

Search any book with either the name of the book or name of the author, get all the details of the book. Searching for an author will get all the books written by the author.

The function is named as `getbook()` in the `libraryfunc.py` file and is defined as...

```
def getbook():
    print('''
        CHOOSE AN OPTION
        1) SEARCH BOOK BY NAME
        2) SEARCH BOOK BY AUTHOR
        3) GO BACK
        ''')
    optgb = int(input('Enter your choice :'))
    if optgb == 1:
        book = input('Enter a book name: ')
        if checkbook(book) == True:
            for i in data:
                if i[1] == book:
                    print(f'Book No.{i[0]}, {i[1]} written by {i[2]}.')
        else:
            print('Book does not exist! Try again!')
            getbook()
    elif optgb == 2:
```

```
        author = input('Enter an author name: ')
        if checkauthor(author) == True:
            for i in data:
                if i[2] == author:
                    print(f'Book No.{i[0]}, {i[1]} written by {i[2]}.')
        else:
            print('Author does not exist! Try again!')
            getbook()
    elif optgb == 3:
        main.libscreen()
```

2.5 CHECK BOOK AVAILABILITY

Check the availibilty of a book with the name of the book from the database.

The function is named as `checkbookavail()` in the `libraryfunc.py` file and is defined as...

```
def checkbookavail(bookno):
    a = 0
    for i in data:
        if i[0]==bookno:
            if i[-1]==1:
                print('Book is Available!')
                a = 1
            else:
                print('Book is not available!')
                a = 0
    return a
```

3. RECORD FUCNTIONS

The `recordfunc.py` file contains all the record functions below. These function work with all the records and the library database.

3.1 BORROW A BOOK

The function will take your **name**, **class** and the **serial number** of the book you want to borrow from the library as the input and add to the record. This function will also change the availability of the borrowed book.

The function to insert a record in the record database `insertrecord()` in `recordfunc.py` file and the fucntion to `borrowbook(book)` in the `libraryfunc.py` and are defiined as...

```
def insertrecord(name,classs,date,bookno):
    cur2.execute('''insert into record values(%d,"%s","%s","%s",%d)'''%
(lenofdata+1,date,name,classs,bookno))
    db2.commit()
```

```
def borrowbook(bookno):
    for i in data:
        if i[0]==bookno:
            cur.execute('''update lib set available="0" where sno="%s"'''%
(bookno))
            db.commit()
            print('UPDATED DATABASE SUCCESFULLY')
```

3.2 RETURN A BOOK

The function will take your **name**, **class** and the **serial number** of the book you want to return from the library as the input and add to the record. This function will also change the availability of the returned book.

The function to insert a record in the record database `insertrecord()` in `recordfunc.py` file and the fucntion to `returnbook(book)` in the `libraryfunc.py` and are defiined as...

```
def insertrecord(name,classs,date,bookno):
    cur2.execute('''insert into record values(%d,"%s","%s","%s",%d)'''%
(lenofdata+1,date,name,classs,bookno))
    db2.commit()
```

```
def returnbook(bookno):
    for i in data:
        if i[0]==bookno:
            cur.execute('''update lib set available="1" where sno="%s"'''%
(bookno))
            db.commit()
            print('UPDATED DATABASE SUCCESFULLY')
```

Hardware and Software

Software used to build this Project

1. JUPYTER NOTEBOOK (6)
2. VS Code (1.74)
3. Python (3.10)
4. MySQL (5.01)

Software required to run this Program

1. Python (Version: 3 or higher)
2. MySQL (Version: 5 or higher)

Hypothesis

This project is practical and can be used in real libraries but with better user friendly interface by including a GUI (Graphical User Interface) and more utility based on the needs of the library. For example the library can charge extra the user for keeping a book for more than a week. No program is ever complete, we can always make them better depending on our needs.

This program will make it easier for libraries to keep a record of all the books in a library and the records of the library.

Content of Syllabus used

1. Python Functions
2. MySQL
3. Python Modules
4. Python MySQL connectivity

Output

Adding a book

```
WELCOME TO THE SCHOOL LIBRARY
CHOOSE AN OPTION
1) THE LIBRARY DATABSE
2) THE LIBRARY RECORDS
3) ABOUT THE LIBRARY MANGEMENT SYSTEM
4) EXIT
Enter you choice: 1
CHOOSE AN OPTION
1) ADD A BOOK
2) DELETE A BOOK
3) UPDATE A BOOK
4) SEARCH A BOOK
5) CHECK BOOK AVAILABILITY
6) GO BACK TO MAIN SCREEN
Enter you choice: 1
Enter the name of the book: Dune
Enter the name of the author: Frank Herbert
Enter the price of the book: 351
Enter the number of pages ini the book: 755
Book added successfully!
DATABASE UPDATED SUCCESSFULLY
```

Deleting a book

```
WELCOME TO THE SCHOOL LIBRARY
CHOOSE AN OPTION
1) THE LIBRARY DATABSE
2) THE LIBRARY RECORDS
3) ABOUT THE LIBRARY MANGEMENT SYSTEM
4) EXIT
Enter you choice: 1
CHOOSE AN OPTION
1) ADD A BOOK
2) DELETE A BOOK
3) UPDATE A BOOK
4) SEARCH A BOOK
5) CHECK BOOK AVAILABILITY
6) GO BACK TO MAIN SCREEN
Enter you choice: 2
Enter the name of the book to delete: Divine Comedy
Book deleted successfully!
DATABASE UPDATED SUCCESSFULLY
```

Updating a book

```
WELCOME TO THE SCHOOL LIBRARY
CHOOSE AN OPTION
1) THE LIBRARY DATABSE
2) THE LIBRARY RECORDS
3) ABOUT THE LIBRARY MANGEMENT SYSTEM
4) EXIT
Enter you choice: 1
CHOOSE AN OPTION
1) ADD A BOOK
2) DELETE A BOOK
3) UPDATE A BOOK
4) SEARCH A BOOK
5) CHECK BOOK AVAILABILITY
6) GO BACK TO MAIN SCREEN
Enter you choice: 3
Enter the name of the book you would like to update: Bisarjan
CHOOSE AN OPTION
1) CHANGE NAME OF THE BOOK
2) CHANGE AUTHOR OF THE BOOK
3) CHANGE NUMBER OF PAGES
4) CHANGE PRICE
Enter your choice: 3
Enter new number of pages: 345
Book updated successfully!
DATABASE UPDATED SUCCESSFULLY
```

Searching a book

```
WELCOME TO THE SCHOOL LIBRARY
CHOOSE AN OPTION
1) THE LIBRARY DATABSE
2) THE LIBRARY RECORDS
3) ABOUT THE LIBRARY MANGEMENT SYSTEM
4) EXIT
Enter you choice: 1
CHOOSE AN OPTION
1) ADD A BOOK
2) DELETE A BOOK
3) UPDATE A BOOK
4) SEARCH A BOOK
5) CHECK BOOK AVAILABILITY
6) GO BACK TO MAIN SCREEN
Enter you choice: 4
CHOOSE AN OPTION
1) SEARCH BOOK BY NAME
2) SEARCH BOOK BY AUTHOR
3) GO BACK
```

```
Enter a book name: My Experiments with Truth
Book No.32, My Experiments with Truth written by Mahatma M.K.Gandhi.
```

Checking availability of a book

```
WELCOME TO THE SCHOOL LIBRARY
CHOOSE AN OPTION
1) THE LIBRARY DATABSE
2) THE LIBRARY RECORDS
3) ABOUT THE LIBRARY MANGEMENT SYSTEM
4) EXIT
Enter you choice: 1
CHOOSE AN OPTION 1) ADD A BOOK 2) DELETE A BOOK 3) UPDATE A BOOK
4) SEARCH A BOOK
5) CHECK BOOK AVAILABILITY 6) GO BACK TO MAIN SCREEN
Enter you choice: 5
Enter book serial number: 31 Book is Available!
```

Borrowing a book

```
WELCOME TO THE SCHOOL LIBRARY
CHOOSE AN OPTION
1) THE LIBRARY DATABSE
2) THE LIBRARY RECORDS
3) ABOUT THE LIBRARY MANGEMENT SYSTEM
4) EXIT
Enter you choice: 2
CHOOSE AN OPTION
1) BORROW A BOOK
2) RETURN A BOOK
3) CHECK BOOK AVAILABILITY 4) BACK TO MAIN PAGE
Enter you choice: 1
ENTER YOUR DETAILS BELOW
Enter your name: Sanchit Saini Enter your class: XII A
Enter book serial number: 31
UPDATED DATABASE SUCCESFULLY
BORROWED BOOK SUCCESSFULLY!
```

Returning a book

```
WELCOME TO THE SCHOOL LIBRARY
CHOOSE AN OPTION
1) THE LIBRARY DATABSE
2) THE LIBRARY RECORDS
3) ABOUT THE LIBRARY MANGEMENT SYSTEM
4) EXIT
```

```
Enter you choice: 2
CHOOSE AN OPTION
1) BORROW A BOOK
2) RETURN A BOOK
3) CHECK BOOK AVAILABILITY 4) BACK TO MAIN PAGE
Enter you choice: 2
ENTER YOUR DETAILS BELOW
Enter your name: Sanchit Saini Enter your class: XII A
Enter book serial number: 31
UPDATED DATABASE SUCCESFULLY
RETURNED BOOK SUCCESSFULLY!
```

Bibliography

1. Websites
    - https://stackoverflow.com/
    - https://www.geeksforgeeks.org/
    - https://www.w3schools.com/
2. Books
    - Computer Science with Python Class 12(Sumita Arora)

Analysis

User Friendly : Highly user friendly with better error handling in both python and MySQL code.

Practical : Can be used in real School libraries to keep a record of all the books and records in the library.