PROJECT ON

LIBRARY MANAGEMENT SYSTEM

Submitted By:

Brahmaiah Rachapudi
Abhishek Nandagawali
Sudhamsa Nagala

INDEX

INTRODUCTION

The aims and objectives are as follows:

- The main aim in library management is adding, issuing and returing of books,and also generating reports on it.

Adding books:

Here we add new books into our database.

Issuing books:

Here we issue books to the candidates.

Returning books:

We get books return from candidates.

Delete books:

We delete books here.

Displaying books:

Here we track all the book details.

Reports:

Here we get information about issued and returned books.

PROJECT ANALYSIS

OPERATION ENVIRONMENT :

FRONT END	PYTHON
DATA BASE	MYSQL

Here we have created a online data base:

To connect to database we have to give,

Host	sql12.freesqldatabase.com
Database name	sql12621331
Database user	sql12621331
Database password	1ssdLqTHym
Port number	3306

FUNCTIONS AND MODULES:

Modules:

- **Import mysql.connector:**

By importing this module,we are able to get the connection between SQL and python.

Functions:

Connect():

It establishes the connection between Python and MYSQL.

Cursor():

It facilitates the row-by-row processing of records in the result set.

Syntax: <cursor object>=<connection object>.cursor()

Execute():

It is used to execute the sql query and get records using python.

Syntax:<cursor object>.execute<sql query string>

Def():

Is a block of code which only runs when it is called.

Fetchall():

Here it will return all the rows from the result set in the form of a tuple containing the records.

Fetchone():

It will return one row from the result set in the form of a tuple containing the records.

Commit():

It provides changes in the database physically.

SYSTEM DESIGN

Table Design:

-Our project has 3 MYSQL tables:-

Book Table for keeping track of book

Field	Data Type	Default	key
bname	Varchar(100)	Not null	
author	Varchar(100)		
bcode	Varchar(10)	Not null	Primary key
total	Int		
Subject	Varchar(50)		

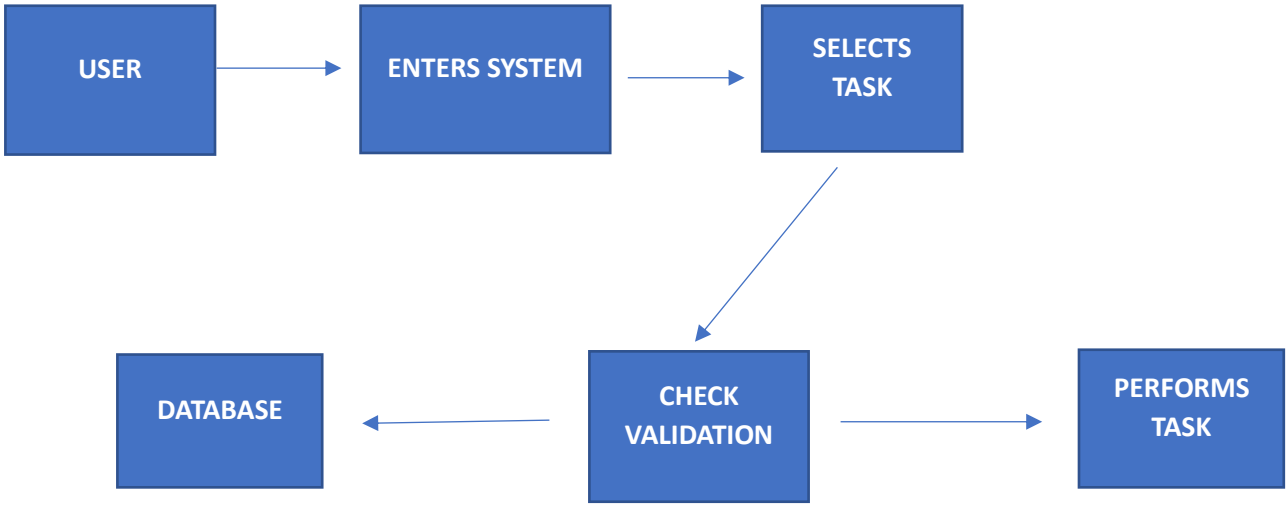
Issue table to issue a book

Field	Data type	Default	Key
name	Varchar(100)		
regno	Int	Not null	Primary key
bcode	Varchar(10)	Not null	
issue_date	Datetime		

Return_books table for returning a book

field	Data type	Default	Key
name	Varchar(100)		
regno	Int	Not null	Primary key
bcode	Varchar(10)	Not null	
Return_date	Datetime		

Data Flow Diagram:



IMPLEMENTATION

Source code:

```
import mysql.connector as a

conn =
a.connect(host='sql12.freesqldatabase.com',user='sql12621331',passw
ord='1ssdLqTHym',database='sql12621331')

my_cursor=conn.cursor()

conn.commit()

print('connection succesfully created')


def addbook():

    book_name = input('enter book name: ')

    book_author = input('enter author name: ')

    book_code = input('enter book code: ')

    total_books = int(input('total books: '))
```

```
    subject = input('enter subject: ')
    data = (book_name, book_author, book_code, total_books, subject)
    sql = 'insert into books values (%s, %s, %s, %s, %s);'
    my_cursor = conn.cursor()
    my_cursor.execute(sql, data)
    conn.commit()
    print('book added successfully')
    wait = input('press enter to continue')

    main()

def issuebook():
    sname = input('enter student name: ')
    reg_no = int(input('enter reg no: '))
    book_code = (input('enter book code: '))
    issue_date = input('enter date: ')
    data = (sname, reg_no, book_code, issue_date)
    sql = 'insert into issue values (%s, %s, %s, %s);'
    my_cursor = conn.cursor()
    my_cursor.execute(sql, data)
    conn.commit()
    print('book issued successfully to:', sname)
```

```
    wait = input('press enter to continue')
    bookupdate(book_code, -1)
    main()

def returnbook():
    sname = input('Enter student name: ')
    reg_no = int(input('Enter reg no: '))
    book_code = (input('Enter book code: '))
    return_date = input('Enter date: ')
    data = (sname, reg_no,book_code, return_date)
    sql = 'insert into return_books values (%s, %s, %s, %s);'
    my_cursor = conn.cursor()
    my_cursor.execute(sql, data)
    conn.commit()
    print('book returned by:',sname)
    wait = input('press enter to continue')
    bookupdate(book_code, 1)

def bookupdate(book_code, update):
    sql_select = 'select total from books WHERE bcode = %s;'
    data = (book_code,)
    my_cursor = conn.cursor()
```

```
    my_cursor.execute(sql_select, data)
    myresult = my_cursor.fetchone()
    t = myresult[0] + update
    sql_update = 'update books SET total = %s WHERE bcode = %s;'
    data=(t,book_code)
    my_cursor.execute(sql_update, data)
    conn.commit()
    wait = input('press enter to continue')
    main()

def deletebook():
    book_code = int(input('enter book code: '))
    sql = 'delete from books WHERE bcode = %s;'
    data = (book_code,)
    my_cursor = conn.cursor()
    my_cursor.execute(sql, data)
    conn.commit()
    print('book deleted successfully')
    wait = input('press enter to continue')
    main()

def displaybook():
```

```
    sql = 'select * from books;'
    my_cursor = conn.cursor()
    my_cursor.execute(sql)
    myresult = my_cursor.fetchall()
    for i in myresult:
        print('book_name:', i[0])
        print('book_author:', i[1])
        print('book_code:', i[2])
        print('total_books:', i[3])
        print('subject:', i[4])
    wait = input('press enter to continue')
    main()

def report_issued_books():
    sql = 'select * from issue;'
    my_cursor = conn.cursor()
    my_cursor.execute(sql)
    myresult = my_cursor.fetchall()
    for i in myresult:
        print(myresult)
    wait = input('press enter to continue')
    main()
```

```
def report_return_books():
    sql = 'select * from return_books;'
    my_cursor = conn.cursor()
    my_cursor.execute(sql)
    myresult = my_cursor.fetchall()
    for i in myresult:
        print(myresult)
    wait = input('press enter to continue')
    main()

def main():
    print('''
    LIBRARY MANAGEMENT SYSTEM APPLICATION
    1. ADD BOOK
    2. ISSUE BOOK
    3. RETURN BOOK
    4. DELETE BOOK
    5. DISPLAY BOOKS
    6. REPORT MENU
    7. EXIT PROGRAM
    ''')
```

```
choice = input('enter task no: ')

if choice == '1':
    addbook()
elif choice == '2':
    issuebook()
elif choice == '3':
    returnbook()
elif choice == '4':
    deletebook()
elif choice == '5':
    displaybook()
elif choice == '6':
    print('''REPORT MENU
    1. ISSUED BOOKS
    2. RETURNED BOOKS
    3. GO BACK TO MAIN MENU
    ''')
    choice = input('enter task no: ')
    if choice == '1':
        report_issued_books()
```

```
        elif choice == '2':
            report_return_books()
        elif choice == '3':
            main()
        else:
            print('please try again')
            main()
    elif choice == '7':
        print('Thank you and have a great day ahead')
    else:
        print('please try again')
        main()

main()
```

System Testing

Here we perform dry testing,

The aim of the system testing process is to determine all defects in our project.

Unit Testing:-

Here we check complete environment i.e. by importing module whether the sql connection has established or not.

Integration Testing:-

Here we check whether we are getting expected outputs are not.

- ✓ To add book, it will check the book code should be unique value.
- ✓ To issue book, before it will validate the student details like student registration number should be unique value.
- ✓ Return book, here validates the book code and student registration number
- ✓ Report on books, validates whether the system updating about book issued and book returned details.

We user selects other than the mention tasks it will display as " please try again".

OUTPUTS AND TABLES

Add Book:

File Edit Selection View Go Run Terminal Help

LMS_project.py

C: > Users > nnage > OneDrive > Desktop > library_management_system > LMS_project.py > ...

```python

def addbook():
    book_name = input('enter book name: ')
    book_author = input('enter author name: ')
    book_code = input('enter book code: ')
    total_books = int(input('total books: '))
    subject = input('enter subject: ')
    data = (book_name, book_author, book_code, total_books, subject)
    sql = 'insert into books values (%s, %s, %s, %s, %s);'
    my_cursor = conn.cursor()
    my_cursor.execute(sql, data)
    conn.commit()
    print('book added successfully')
    wait = input('press enter to continue')
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
    LIBRARY MANAGEMENT SYSTEM APPLICATION
    1. ADD BOOK
    2. ISSUE BOOK
    3. RETURN BOOK
    4. DELETE BOOK
    5. DISPLAY BOOKS
    6. REPORT MENU
    7. EXIT PROGRAM

enter task no: 1
enter book name: Microprocessors and Interfacing
enter author name: N.Senthil kumar
enter book code: A001
total books: 10
enter subject: Electronics
book added successfully
press enter to continue
```

Display Books:

File Edit Selection View Go Run Terminal Help LMS_project

LMS_project.py

C: > Users > nnage > OneDrive > Desktop > library_management_system > LMS_project.py > issuebook

```python
def displaybook():
    sql = 'select * from books;'
    my_cursor = conn.cursor()
    my_cursor.execute(sql)
    myresult = my_cursor.fetchall()
    for i in myresult:
        print('book_name:', i[0])
        print('book_author:', i[1])
        print('book_code:', i[2])
        print('total_books:', i[3])
        print('subject:', i[4])
        print('\n\n')
    wait = input('press enter to continue')
    main()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
book_author: syed m
book_code: 10
total_books: 10
subject: computers



book_name: global warming
book_author: stephen
book_code: b020
total_books: 10
subject: science



book_name: the china wall
book_author: dinesh mukharjee
book_code: b050
total_books: 1
subject: economy



book_name: Microprocessors and Interfacing
book_author: N.Senthil kumar
book_code: A001
total_books: 9
subject: Electronics



press enter to continue
```

Issue Book:

File Edit Selection View Go Run Terminal Help

LMS_project.py

C: > Users > nnage > OneDrive > Desktop > library_management_system > LMS_proj

```python
def issuebook():
    sname = input('enter student name: ')
    reg_no = int(input('enter reg no: '))
    book_code = (input('enter book code: '))
    issue_date = input('enter date: ')
    data = (sname, reg_no, book_code, issue_date)
    sql = 'insert into issue values (%s, %s, %s, %s);'
    my_cursor = conn.cursor()
    my_cursor.execute(sql, data)
    conn.commit()
    print('book issued successfully to:', sname)
    wait = input('press enter to continue')
    bookupdate(book_code, -1)
    main()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
    LIBRARY MANAGEMENT SYSTEM APPLICATION
    1. ADD BOOK
    2. ISSUE BOOK
    3. RETURN BOOK
    4. DELETE BOOK
    5. DISPLAY BOOKS
    6. REPORT MENU
    7. EXIT PROGRAM

enter task no: 2
enter student name: sudha
enter reg no: 537
enter book code: A001
enter date: 2023-05-28
book issued successfully to: sudha
press enter to continue
```

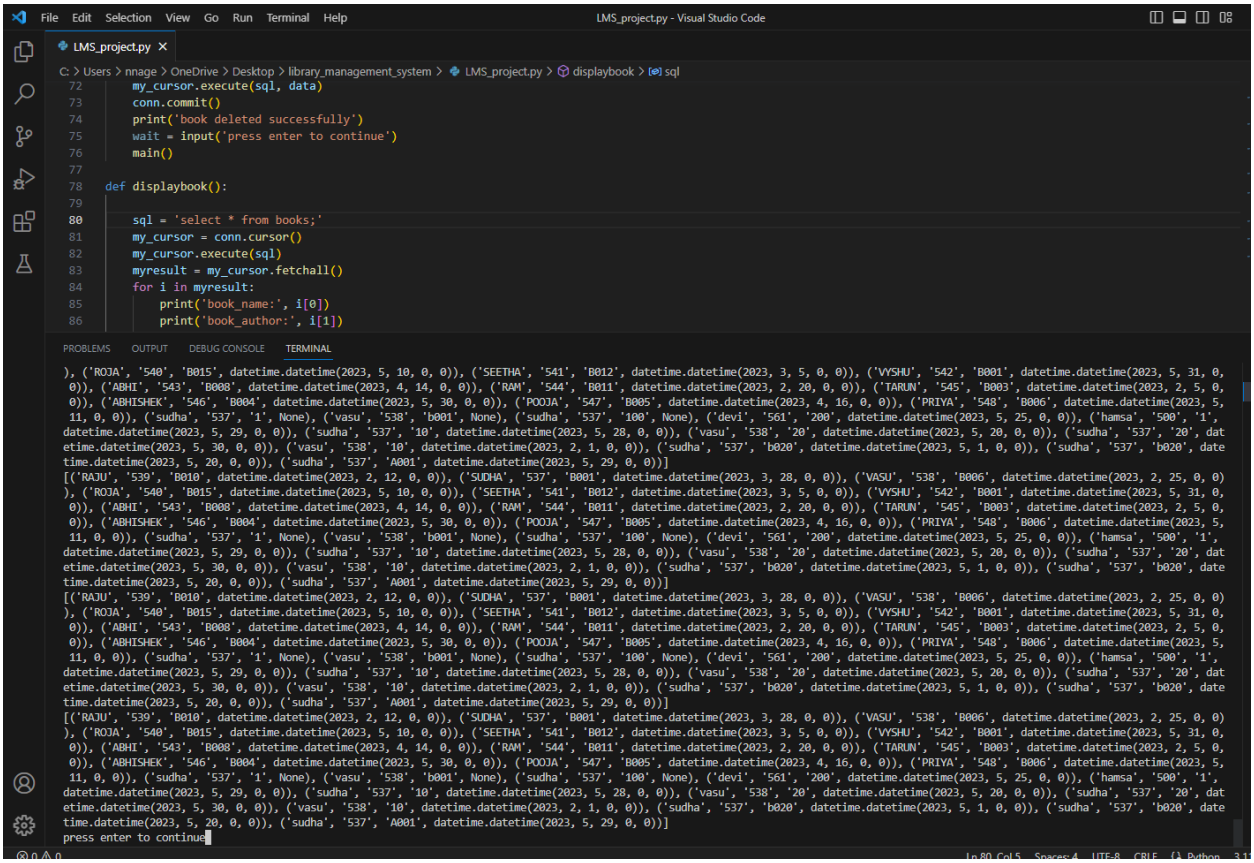
Return Book:

```
press enter to continue

    LIBRARY MANAGEMENT SYSTEM APPLICATION
    1. ADD BOOK
    2. ISSUE BOOK
    3. RETURN BOOK
    4. DELETE BOOK
    5. DISPLAY BOOKS
    6. REPORT MENU
    7. EXIT PROGRAM

enter task no: 3
Enter student name: sudha
Enter reg no: 537
Enter book code: A001
Enter date: 2023-05-29
book returned by: sudha
press enter to continue
```

⊗ 0 ⚠ 0

86°F
Haze

Returned Book_Report:



Delete Book:

File Edit Selection View Go Run Terminal Help

LMS_project.py

C: > Users > nnage > OneDrive > Desktop > library_management_system >

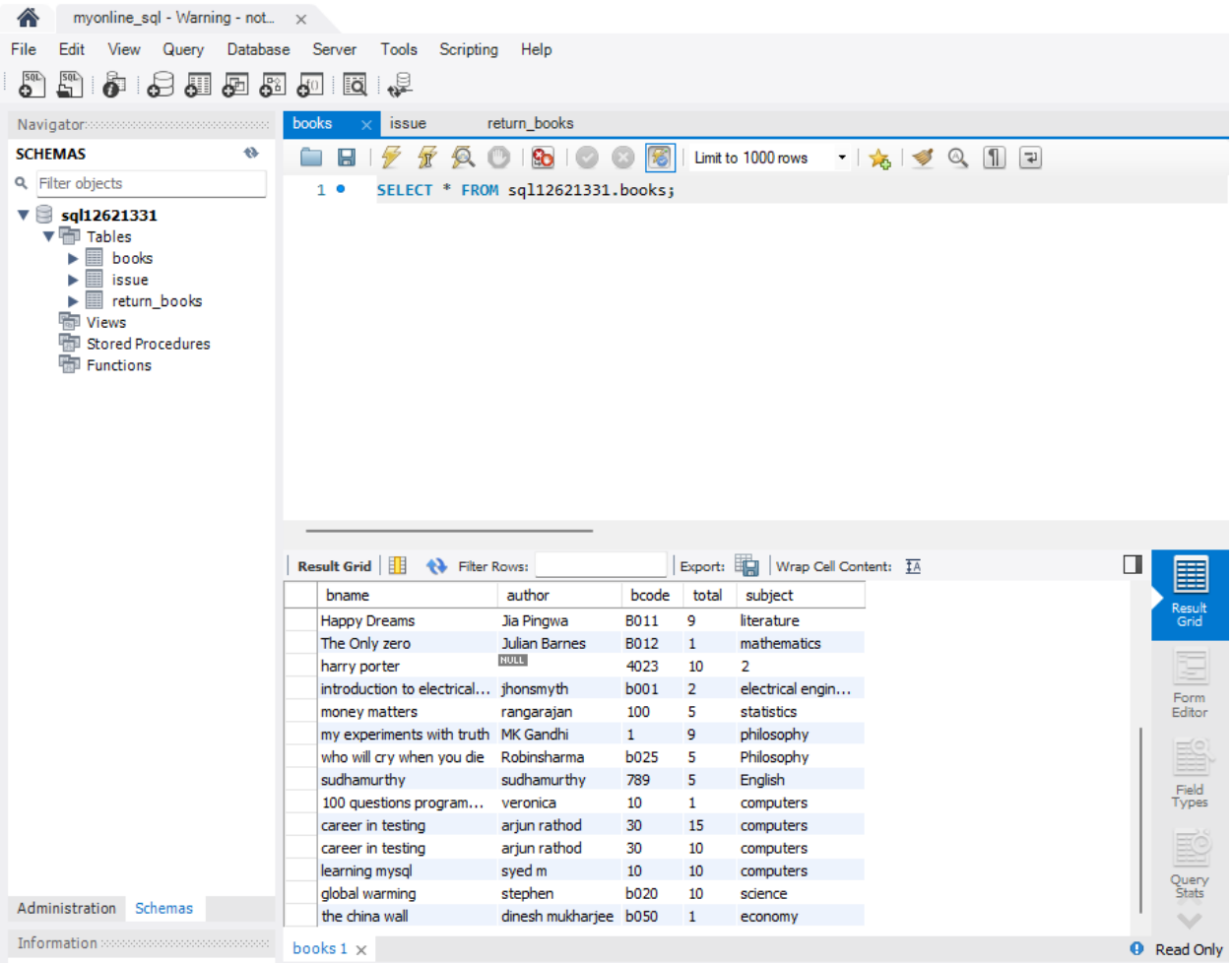
```python
    main()

def deletebook():
    book_code = input('enter book code: ')
    sql = 'delete from books WHERE bcode = %s;'
    data = (book_code,)
    my_cursor = conn.cursor()
    my_cursor.execute(sql, data)
    conn.commit()
    print('book deleted successfully')
    wait = input('press enter to continue')
    main()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
    LIBRARY MANAGEMENT SYSTEM APPLICATION
    1. ADD BOOK
    2. ISSUE BOOK
    3. RETURN BOOK
    4. DELETE BOOK
    5. DISPLAY BOOKS
    6. REPORT MENU
    7. EXIT PROGRAM

enter task no: 4
enter book code: A001
book deleted successfully
press enter to continue
```

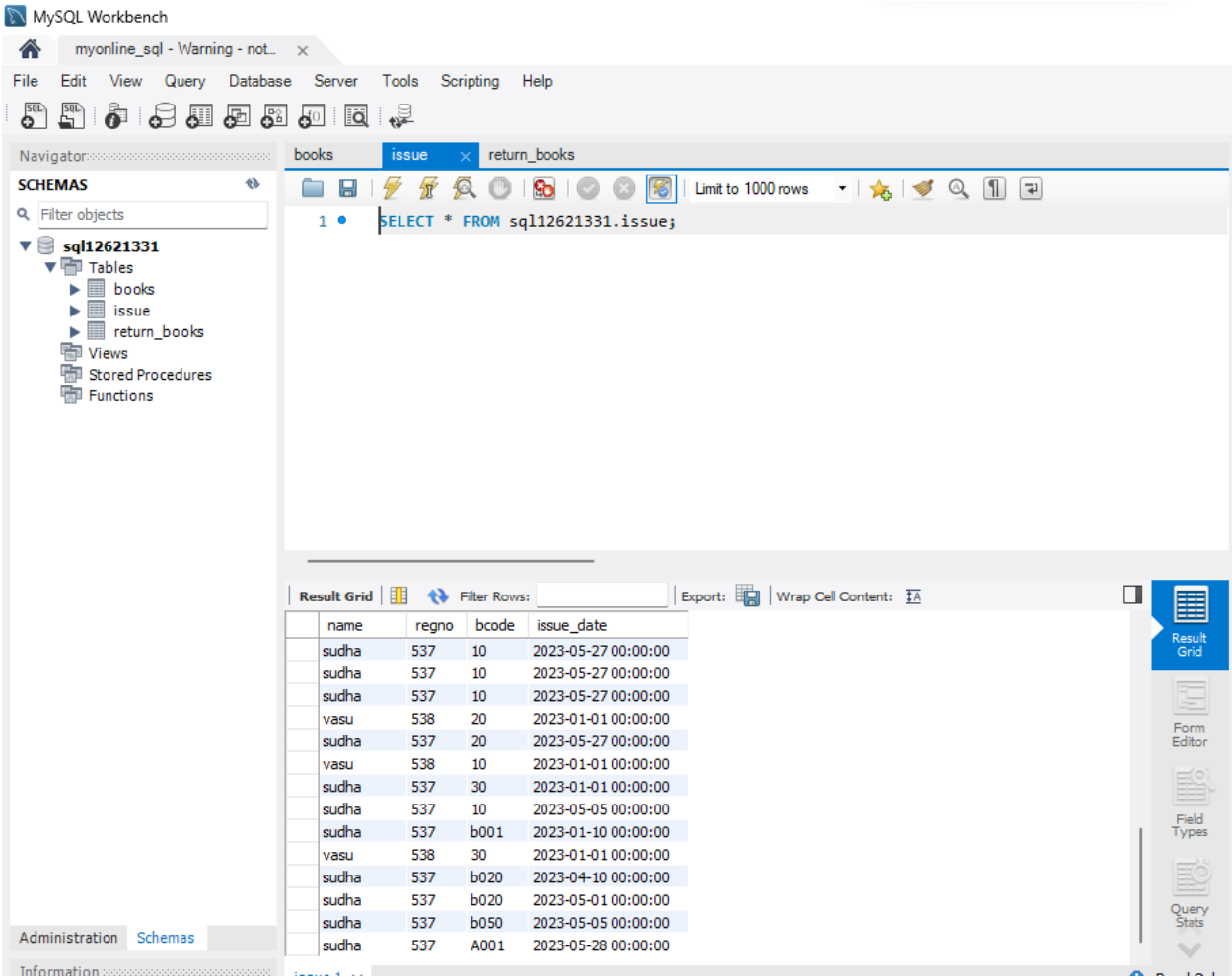
SQL Tables:

Book Table:



bname	author	bcode	total	subject
Happy Dreams	Jia Pingwa	B011	9	literature
The Only zero	Julian Barnes	B012	1	mathematics
harry porter	NULL	4023	10	2
introduction to electrical...	jhonsmyth	b001	2	electrical engin...
money matters	rangarajan	100	5	statistics
my experiments with truth	MK Gandhi	1	9	philosophy
who will cry when you die	Robinsharma	b025	5	Philosophy
sudhamurthy	sudhamurthy	789	5	English
100 questions program...	veronica	10	1	computers
career in testing	arjun rathod	30	15	computers
career in testing	arjun rathod	30	10	computers
learning mysql	syed m	10	10	computers
global warming	stephen	b020	10	science
the china wall	dinesh mukharjee	b050	1	economy

Issue Table:



name	regno	bcode	issue_date
sudha	537	10	2023-05-27 00:00:00
sudha	537	10	2023-05-27 00:00:00
sudha	537	10	2023-05-27 00:00:00
vasu	538	20	2023-01-01 00:00:00
sudha	537	20	2023-05-27 00:00:00
vasu	538	10	2023-01-01 00:00:00
sudha	537	30	2023-01-01 00:00:00
sudha	537	10	2023-05-05 00:00:00
sudha	537	b001	2023-01-10 00:00:00
vasu	538	30	2023-01-01 00:00:00
sudha	537	b020	2023-04-10 00:00:00
sudha	537	b020	2023-05-01 00:00:00
sudha	537	b050	2023-05-05 00:00:00
sudha	537	A001	2023-05-28 00:00:00

Return Table: