**MINOR PROJECT REPORT**

**On**

**LIBRARY MANAGEMENT SOFTWARE**

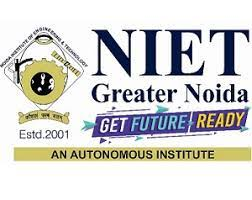
Submitted in Partial fulfilment for the Award of the degree of

**Bachelor of Technology**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

Submitted to



**ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW (U.P)**

**Under the guidance of Submitted By**

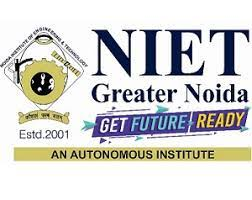
**Mr. Ibrar DEEPANSHU BHATI-(****2101330100093)**

**(Assistant professor) AYUSHI - (2101330100086)**

**Mr.** **ASHISH MATHUR ASUTOSH NEGI - (2101330100076)**

**(Assistant professor)**

|  |  |  |
| --- | --- | --- |
| **TABLE OF CONTENTS** | | |
|  |  | **Page No.** |
| **CANDIDATE’S DECLARATION** |  | i |
| **CERIFICATE OF INSTITUTE** |  | ii |
| **ACKNOWLEDGEMENT**  **ABSTRACT** |  | Iii  iv |
| **Table of Contents** |  | v |
| **CHAPTER 1** | **INTRODUCTION** | **7-8** |
|  | 1.1 Overview and Motivation | 7 |
|  | 1.2 Objective | 7 |
|  | 1.3 Issues and Challenges | 7 |
|  | 1.4 Contribution  1.5 Organisation of the Project Report | 7  8 |
| **CHAPTER 2** | **Software Requirement** | **9-19** |
|  | 2.1 Hardware and Software Requirements | 9 |
|  | 2.2 Library Management Software | 10 |
|  | 2.3 Tools and Languages |  |
|  | 2.3.1 Vs code | 12 |
|  | 2.3.2 Python | 13 |
|  | 2.3.3 Tkinter | 14 |
|  | 2.3.3 Ttkthemes | 15 |
|  | 2.3.3 Basic Terminology | 16 |
|  | 2.3.3 System Model | 17 |
|  | 2.3.3 Database Model | 18 |
|  | 2.3.3 Use Case Diagram | 19 |
| **CHAPTER 3** | **Code Snippets** | **20-30** |
| **CHAPTER 4** | **Software Screenshots** | **31-36** |



* **NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY,**
* **GREATER NOIDA**

Approved by AICTE, New Delhi & Govt. of U.P. Affiliated to Abdul Kalam Technical University, Lucknow

NIET Campus, Knowledge Park-2, Gr. Noida-201310 (UP) INDIA

----------------------------------------------------------------------------------------------------------------

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CANDIDATE’S DECLARATION**

I hereby declare that the Minor Project report on “Library Management Software Using Python”

which is being presented here for the partial fulfilment of the requirement of Degree of “Bachelor of Technology ” has been carried out at Department of Computer Science & Engineering, Noida Institute of Engineering & Technology. The technical information provided in this report is presented with due permission of the authorities from the training organization.

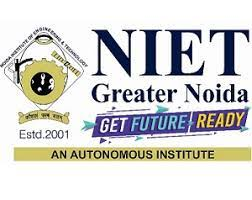
**Deepanshu Bhati (2101330100093)**

**Ayushi (2101330100093)**

**Ashutosh Negi (2101330100093)**

Phone No.- +91-801 050 0700

E-mail: director@niet.co.in   
Website: <https://www.niet.co.in/>



* **NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY,**
* **GREATER NOIDA**

Approved by AICTE, New Delhi & Govt. of U.P. Affiliated to Abdul Kalam Technical University , Lucknow

NIET Campus, Knowledge Park-2, Gr. Noida-201310 (UP) INDIA

----------------------------------------------------------------------------------------------------------------

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CERIFICATE OF INSTITUTE**

This is to certify that Mr. Deepanshu Bhati (2101330100093), Ayushi (2101330100093), Ashutosh Negi (2101330100093) of B.Tech/M.Tech Department Computer Science & Engineering has successfully **completed** his Minor Project during the academic year 2023-2024 as partial fulfilment of the Bachelor of Technology in Information Technology.

Mr. IBRAR &

Mr. ASHISH MATHUR

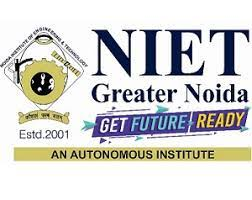
Faculty Coordinator

Computer Science & Technology

KUMUD SAXENA

Head of Department

Computer Science & Engineering



* **NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY,**
* **GREATER NOIDA**

Approved by AICTE, New Delhi & Govt. of U.P. Affiliated to Abdul Kalam Technical University , Lucknow

NIET Campus, Knowledge Park-2, Gr. Noida-201310 (UP) INDIA

----------------------------------------------------------------------------------------------------------------

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**ACKNOWLEDGEMENT**

We are thankful to all individuals who have lent us support and guidance without which we could not have completed our project in stipulated period of time .First and foremost we would like to express our deepest gratitude to our project guide Mr. Ashish Mathur, Department of Computer Science and Engineering for his invaluable support, guidance, motivation, encouragement throughout the period .His incisive inputs, his concern and assistant even during simulation have been extremely helpful. It was their enthusiastic and progressive outlook towards the project which inspired us throughout our work during the period and was always available for guidance. We are grateful to Staff member of Computer Science & Technology department ,for their valuable suggestions and input during the course of the project work .We also extend our thanks to our fellow students for their friendly cooperation.

**1.** **Deepanshu Bhati**

**2.** **Ayushi**

**3.** **Ashutosh Negi**

This is to certify that the above statement made by the candidate is correct to the best of my possible.

ABSTRACT

In this project, we as a group are solving the problem which occurs in offline Library management systems, This Library Management Software is an offline software which make library management much easier with the help of technology.

The main aim of this project is to save time of people by providing easy and quick solution. The offline library management software is specifically designed to manage the day-to-day operations of a library. The software includes features such as issuing and returning of books, managing user details, book details, availability, and the number of copies. The system also includes a fine management module that enables the m The offline library management software is specifically designed to manage the day-to-day operations of a library. The software includes features such as issuing and returning of books, managing user details, book details, availability, and the number of copies. The system also includes a fine management module that enables the management of fines for late returning of books. To access any of these features, login credentials are required to ensure security. Overall, this library management software streamlines the management of libraries by automating various processes and improving efficiency.

Chapter Formation for Research Based Project

**Chapter 1: Introduction**

Our project is an offline library management software that automates various library processes and improves efficiency.

**1.1Overview and Motivation**

Our offline library management software is designed to streamline the day-to-day operations of a library by automating various processes and improving efficiency. The motivation behind this project was to create a comprehensive system that would make it easier for librarians to manage their collections and provide better service to library patrons. By simplifying tasks such as book checkouts, returns, and fine management, we hope to reduce the workload on library staff and improve the overall user experience for library patrons.

1.2Objective

The main aim of this project is to streamline the day-to-day operations of a library, automate various processes, and improve efficiency.

1.3Issues and Challenges

There were multiple issues and challenges that were faced during development of the project like Technical Challenges, Data Accuracy and Security, Compatibility, User Experience, Maintenance and Upgradation.

1.4Contribution

This application contributes to the field of library management by providing an offline software that automates various processes such as book issuing and returning, user and book details management, availability tracking, and fine management.

1.5Organisation of the Project Report

The Organization of the Project Report should provide a clear and concise overview of the structure, scope, and goals of the project. It should include sections like Introduction, Project Design and Methodology, Literature Review, Results, Discussion, Conclusion, References. Overall, the Organization of the Project Report should be well-structured, clearly written, and provide a comprehensive overview of the project and its outcomes.

Chapter 2: Software Requirement

2.1 HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirement: -

• Processor: intel i3 or higher (must have at least 2 GHz clock speed)

• Operating System: Windows, Mac Os, Linux

• RAM: 4 GB (or higher)

• Hard Disk: 10 GB (or higher)

Software Requirement: -

• Programming Language: Python

• Database Management System: CSV

• Text Editor/IDE: Vs Code or Python IDLE

• Libraries/Frameworks: Tkinter, ttkthemes

2.2 Library Management Software

The offline library management software is a computer program designed to manage the day-to-day operations of a library. It includes features such as issuing and returning of books, managing user details, book details, availability, and the number of copies. The system also includes a fine management module that enables the management of fines for late returning of books.

From the perspective of the three students who developed this software, the library management software is a tool that helps to automate several processes in a library, making it easier to manage various operations efficiently. The software has been designed with a user-friendly interface, making it easy for librarians to navigate and use.

The software’s most essential features include the ability to issue and return books, which are crucial functionalities that ensure that the library's inventory remains up to date. With the help of this feature, librarians can keep track of the number of books borrowed by each user and the number of copies available.

The management of user details is another critical feature of the software. This functionality allows librarians to maintain accurate records of users' information, including their names, addresses, and contact information. By keeping updated records of users, the library can easily contact them when necessary.

Additionally, the library management software helps librarians keep track of the availability of books. They can see how many copies of a particular title are available, how long they've been borrowed, and when they're due for return. This feature ensures that the library never runs out of stock of any book.

Finally, the software includes a fine management module that helps librarians manage fines for late returns. Late returns can result in inconvenience for other library users, so this feature helps to incentivize timely returns.

Overall, the library management software is an essential tool for libraries, helping to streamline and automate various processes, improving efficiency, and providing accurate records of users and books.

**2.3Tools and Languages**

**VISUAL STUDIO CODE:** Visual Studio Code is a source-code editor made by Microsoft for Windows, Linux and mac OS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

**Python:** Python is a high-level, interpreted programming language that was first released in 1991. It is an open-source language, meaning that its source code is freely available for anyone to modify and distribute. Python's syntax is designed to be easy to read and write, making it an excellent language for beginners to learn.

Python supports multiple programming paradigms, including object-oriented, functional, and procedural programming. It also has a vast standard library that provides many tools for performing common tasks, such as working with files, networking, and regular expressions. Furthermore, Python supports third-party libraries, which can extend its functionality even further.

One of the distinctive features of Python is its use of whitespace to delimit code blocks, rather than curly braces or other characters like semicolons. This feature helps to make Python code more readable and helps enforce coding standards.

Python is widely used in various domains, including web development, scientific computing, data analysis, artificial intelligence, and machine learning. Its popularity has grown significantly over the years, and today it is one of the most popular programming languages around the world.

Overall, Python is a powerful and versatile programming language that is easy to learn and use. Its focus on readability, simplicity, and flexibility makes it an attractive option for developers across various domains.

**Tkinter:** Tkinter is a built-in Python library used for creating graphical user interfaces (GUIs). It provides a set of tools and widgets that allow developers to create windows, buttons, labels, text fields, and other GUI elements for their applications.

Tkinter is based on the Tcl/Tk GUI toolkit and is included with most Python installations. It is a cross-platform library that works on Windows, macOS, and Linux, making it ideal for developing applications that need to run on multiple platforms.

Tkinter's API is relatively straightforward and easy to learn. Its most basic element is a widget, which is a graphical component that can be placed onto a window or frame. Tkinter includes various types of widgets, including button widgets, label widgets, entry widgets, text widgets, canvas widgets, and more.

Developers can customize the appearance of these widgets by changing their size, color, font, and other properties. Tkinter also allows developers to bind functions to events, such as button clicks, so they can specify what happens when the user interacts with a widget.

Overall, Tkinter is a powerful tool for building simple to complex GUI applications in Python. It provides developers with an extensive set of tools and widgets that enable them to design beautiful and interactive applications quickly.

**Ttkthemes:** ttkthemes is a Python library built on top of Tkinter's ttk module that provides an extensive set of themes for creating beautiful and modern-looking GUI applications. It allows developers to easily customize the appearance of their applications, including colors, fonts, styles, and other graphical elements.

The library includes several pre-built themes, such as "Arc", "Black", "Blue", "Classic", "Equilux", "Plastik", "Radiance", "Scidblue", "Smog", "Ubuntu", and more. Each theme has a unique look and feel, enabling developers to choose the one that best fits their application's design.

ttkthemes also provides tools for creating custom themes, allowing developers to create their own unique styles from scratch or modify existing ones. Additionally, it supports loading themes from external files, making it easier to switch between predefined themes without altering the code.

One of the significant advantages of ttkthemes over the standard Tkinter library is its ability to have themed widgets that can adapt to the theme selected by the user. This feature enables developers to build applications that can change their appearance based on the user's preferences, providing a better user experience.

Overall, ttkthemes is a powerful tool for developers looking to create modern-looking GUI applications in Python. It provides an extensive set of pre-built themes, customization tools, and support for loading external themes.

BASIC TERMINOLOGY

**Library Management Software** – A computer program designed to manage and automate various processes in a library efficiently.

**Graphical User Interface (GUI)** – A user interface that allows users to interact with a computer program using graphical elements such as buttons, menus, and icons.

**Tkinter** – A built-in Python library used for creating GUIs.

**Widgets** – Graphical components, such as buttons, labels, text fields, and other GUI elements, that can be placed onto a window or frame.

**Themes** – Predefined sets of colours, fonts, styles, and other graphical elements used to customize the appearance of a GUI application.

**Fine Management Module** – A module that enables the management of fines for late returning of books.

**Object-Oriented Programming (OOP)** – A programming paradigm that uses objects to represent data and behaviour.

**Syntax** – The rules governing the structure of code in a programming language.

**Standard Library** – A collection of modules and tools included with Python to perform common tasks.

**Cross-Platform** – The ability of a program or application to run on multiple operating systems or platforms.

SYSTEM MODEL:

+----------------+ +----------------+ +-----------------------+

| | | | | |

| User | --> | Front-end | --> | Middleware |

| | | Interface | | |

+----------------+ +----------------+ +-----------------------+

^ ^

| |

v v

+-----------------------+ +--------------------+ +---------------------+

| | | | | |

| Book Database | | User Database | | Fines Database |

| | | | | |

+-----------------------+ +---------------------+ +---------------------+

DATABASE MODEL:

+----------+ +----------------+

| Books | | Borrowed Books |

+----------+ +----------------+

^ ^

| |

+--------------+ +-------------------+

| Members | | Books Issued to |

+--------------+ +-------------------+

|  |
| --- |
|  |
| **Column Name** | **Data Type** | **Key** |
| BookID | INT | Primary Key |
| Title | VARCHAR(255) |  |
| Author | VARCHAR(255) |  |
| Publisher | VARCHAR(255) |  |
| PublicationDate | DATE |  |
| ISBN | VARCHAR(13) |  |
|  |
| **Column Name** | **Data Type** | **Key** |
| BorrowID | INT | Primary Key |
| MemberID | INT | Foreign Key |
| BookID | INT | Foreign Key |
| DateIssued | DATE |  |
| DueDate | DATE |  |
|  |
| **Column Name** | **Data Type** | **Key** |
| MemberID | INT | Primary Key |
| Name | VARCHAR(255) |  |

USE CASE DIAGRAM:

User

|

-------------------

| |

View Reports Generate Reports

Feature Feature

Here we have two use cases: "View Reports" and "Generate Reports". These use cases represent specific features of the software system. The "User" actor interacts with the system through these features.

The "View Reports" feature allows the user to view pre-generated reports. This could involve selecting a report from a menu or dashboard and then viewing its contents in the application interface.

The "Generate Reports" feature, on the other hand, allows the user to create new reports based on custom criteria. This might involve selecting data sources, defining filters or query parameters, and specifying how the report should be formatted or presented.

CODE SNNIPPETS:

import libraries



defining some global vars



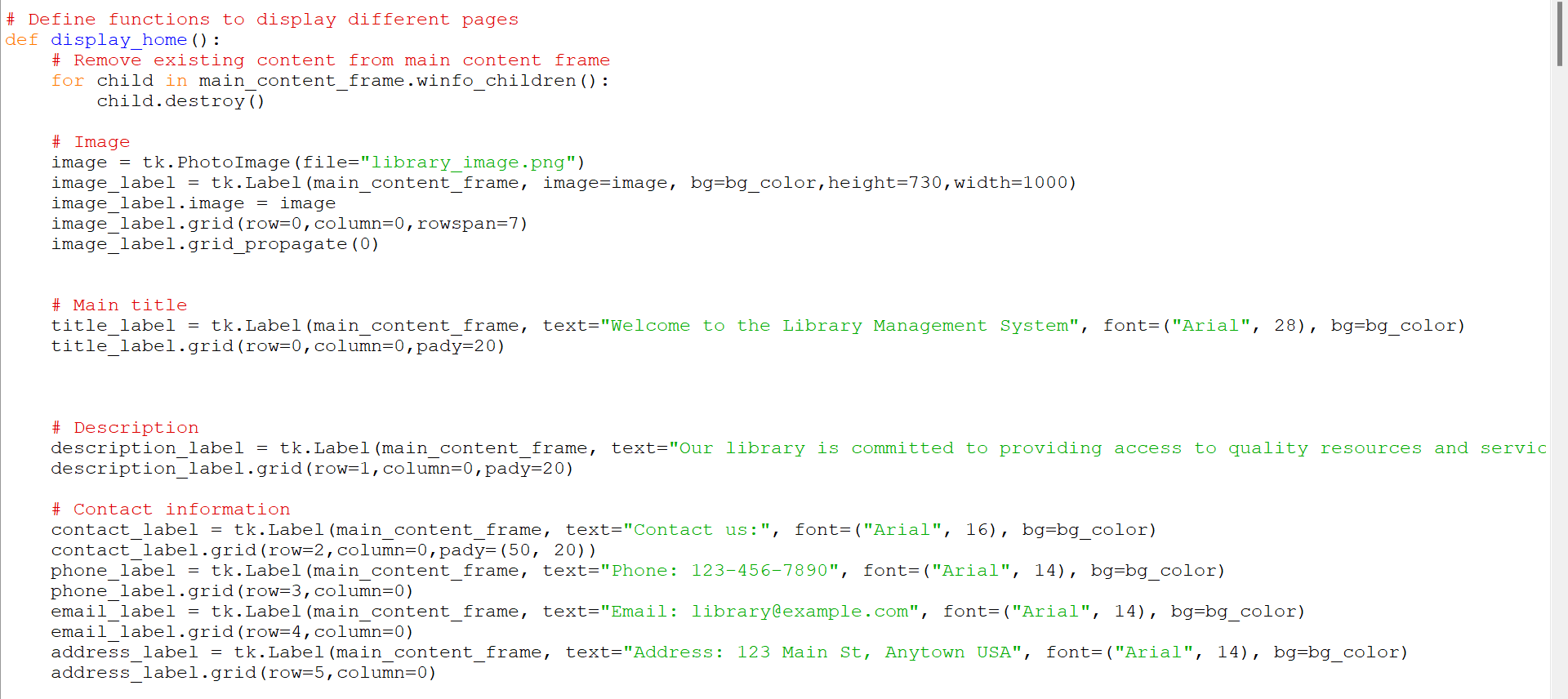
setting variables for colours



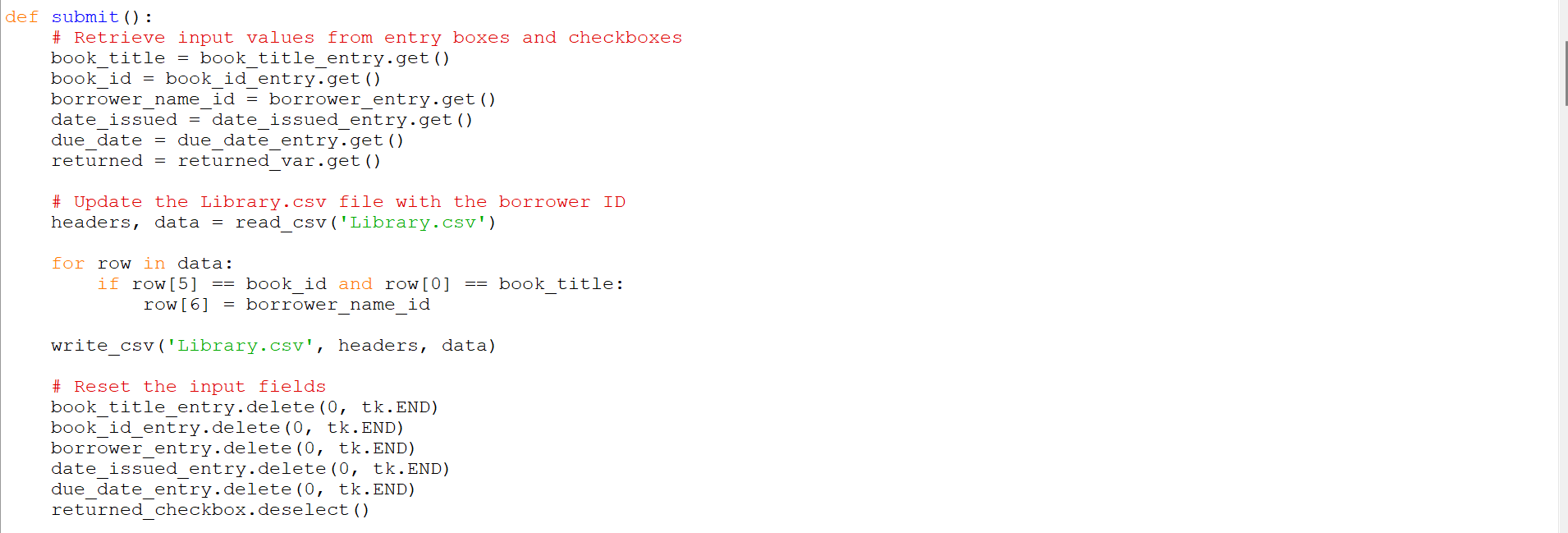
read and write csv function



display home function



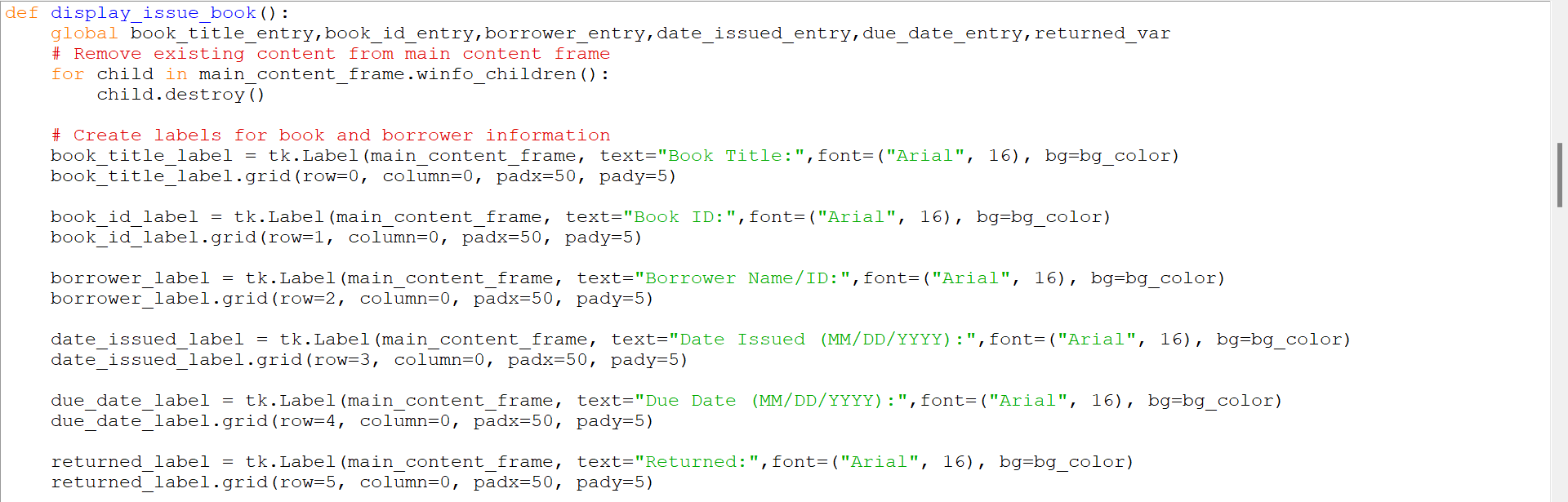
Submit function



cancel function

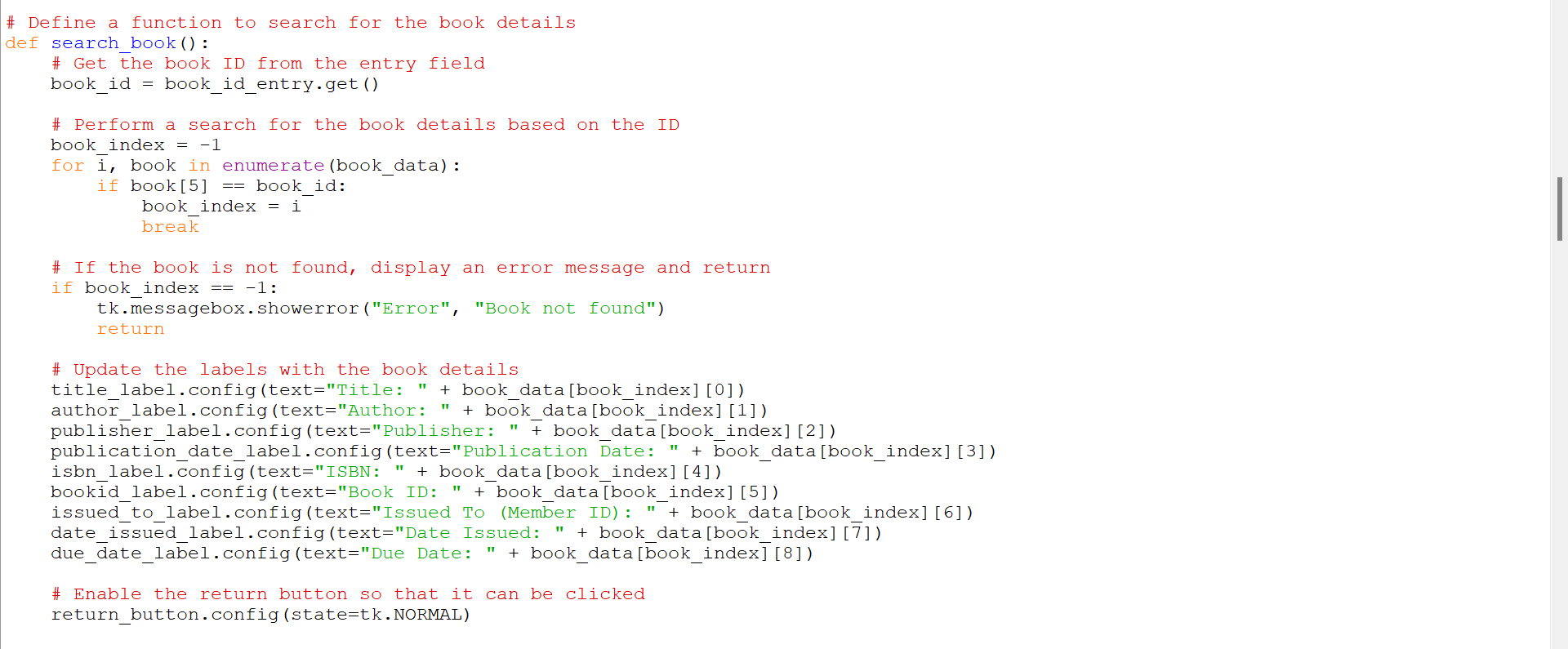


display issue books function

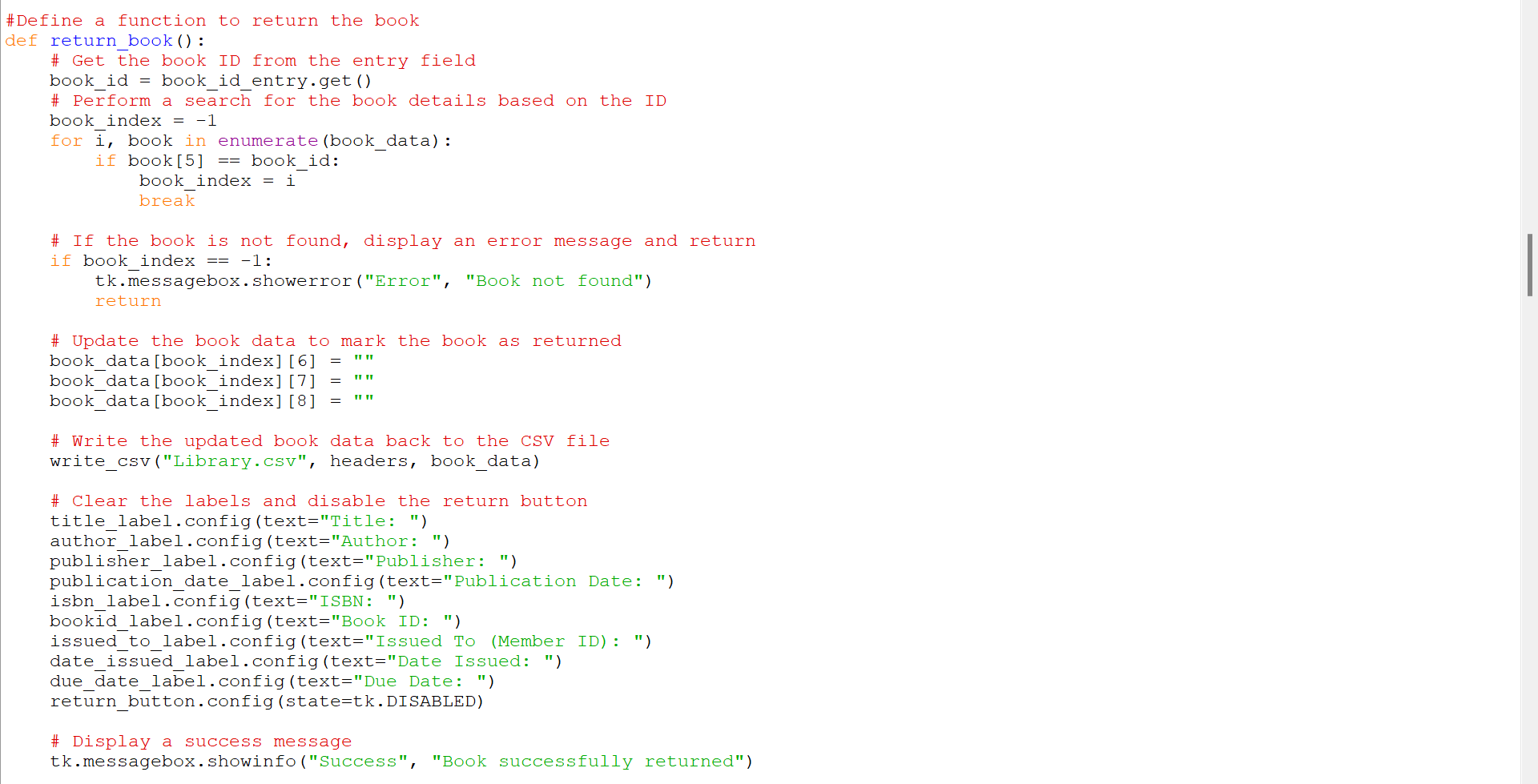




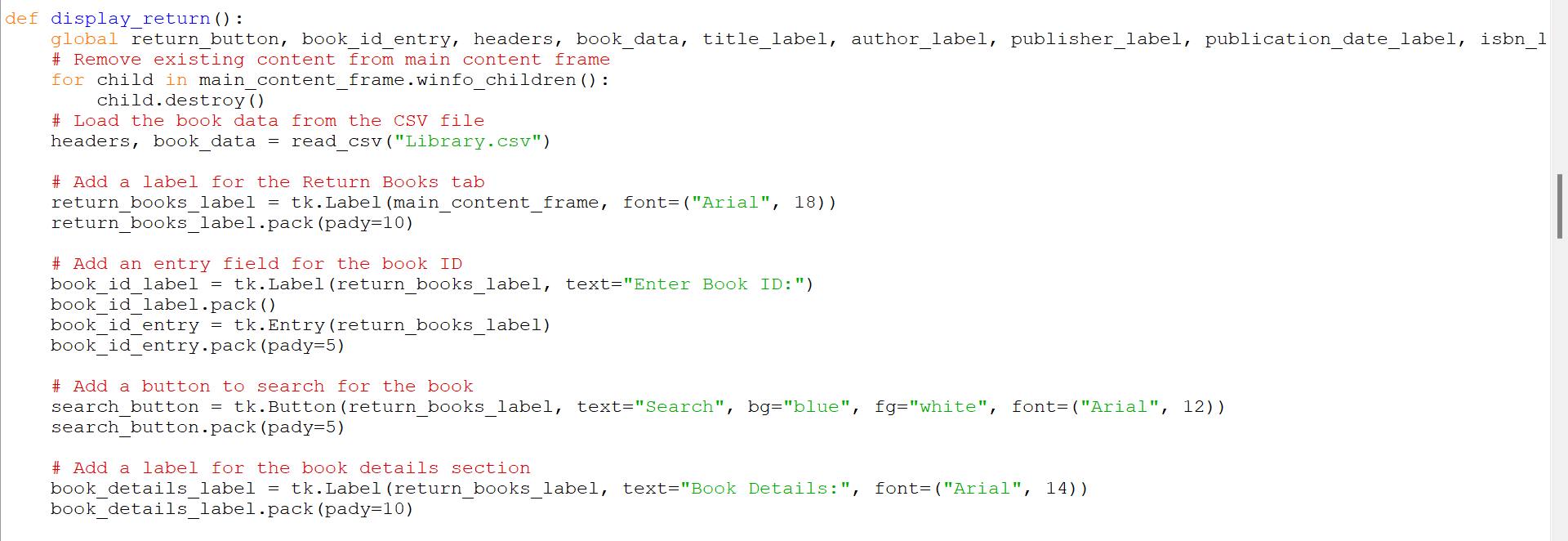
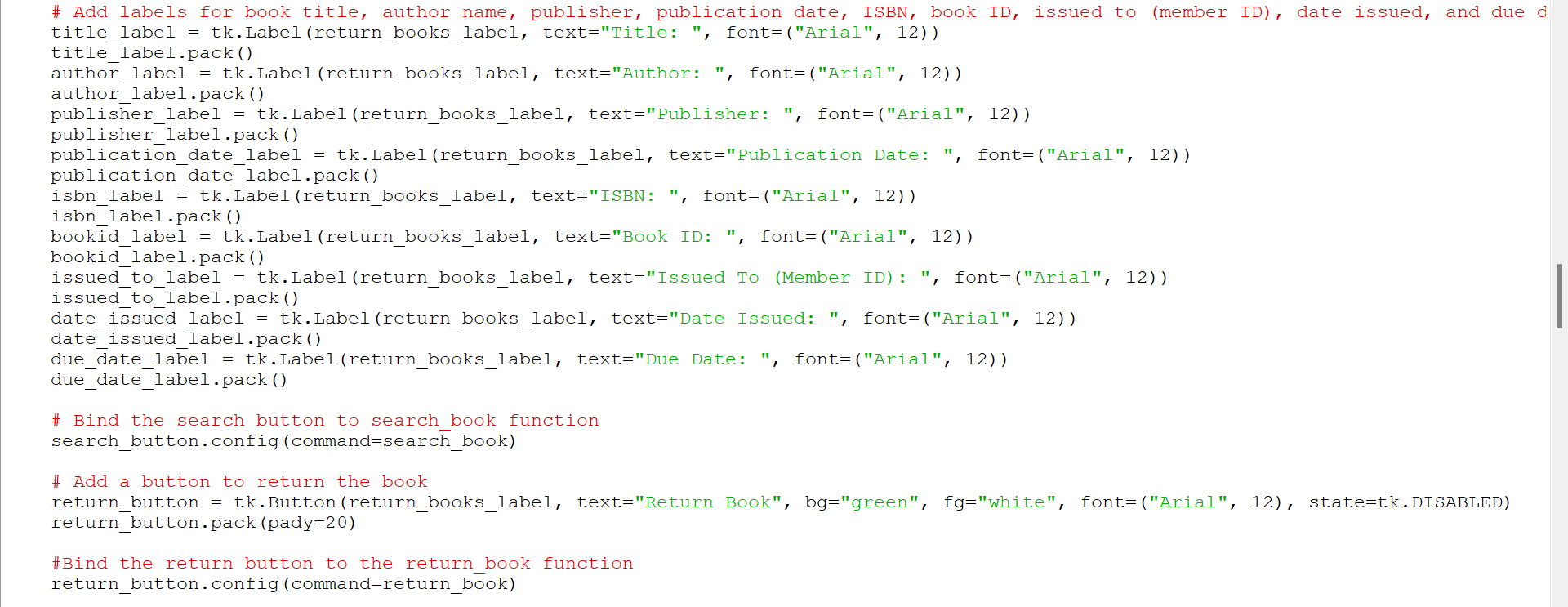
search book function



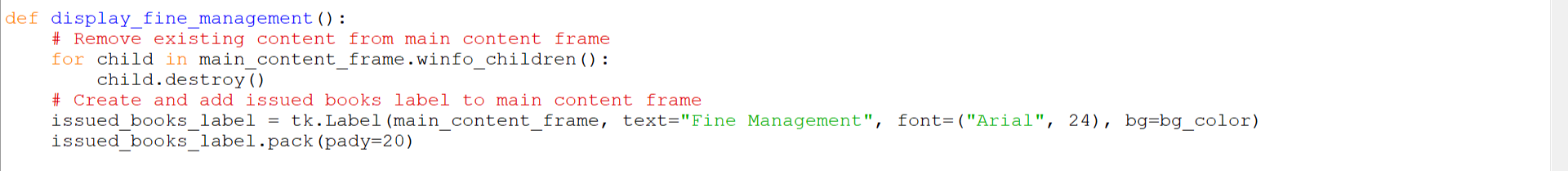
return book function



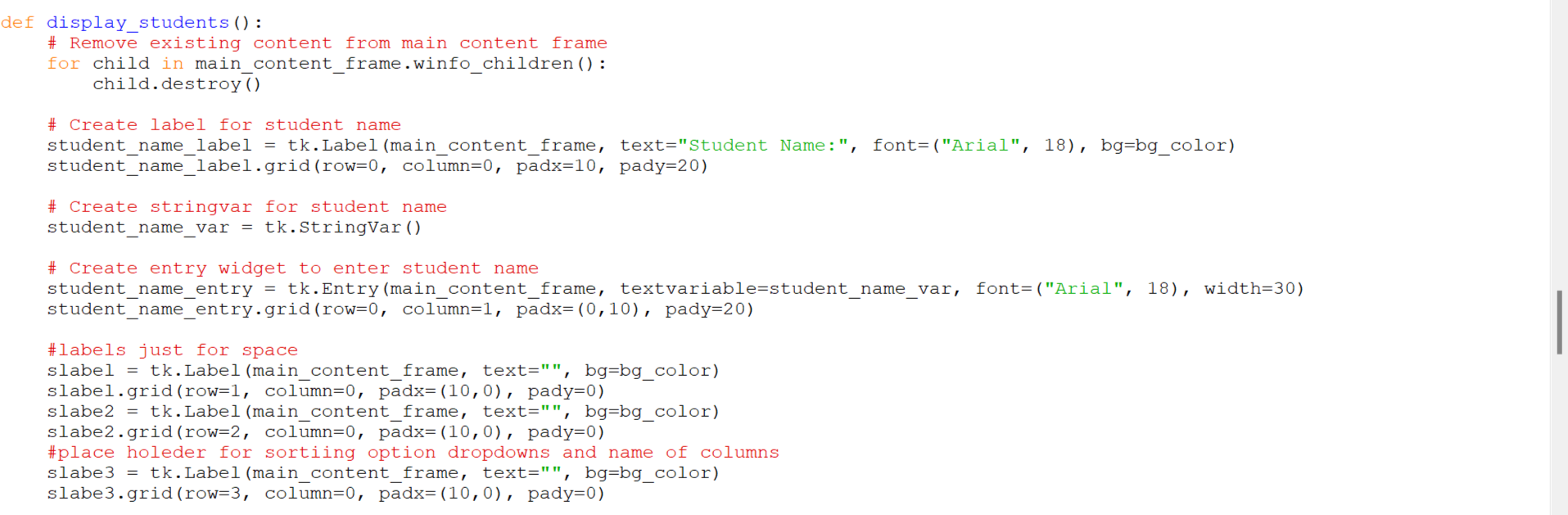
display return function



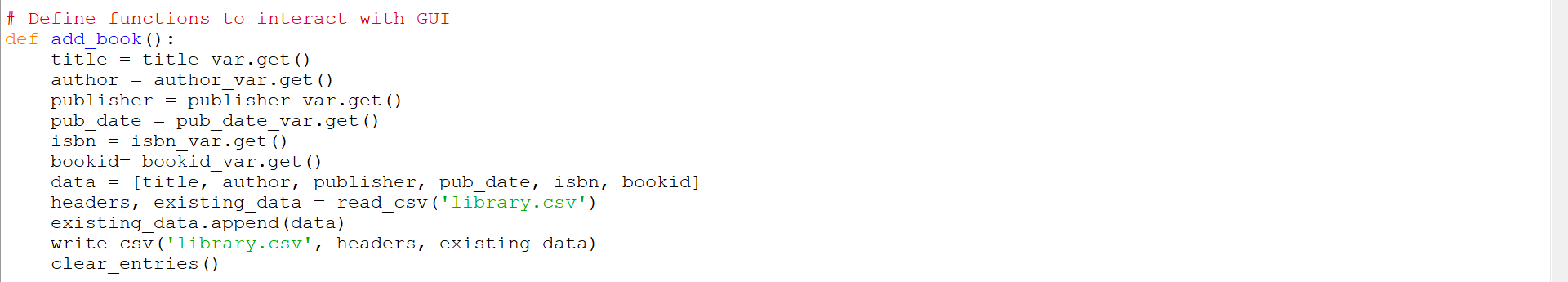
display fine management function



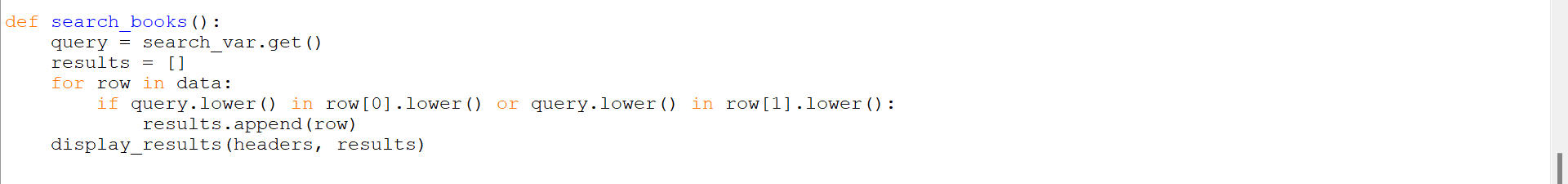
display students function



add book function

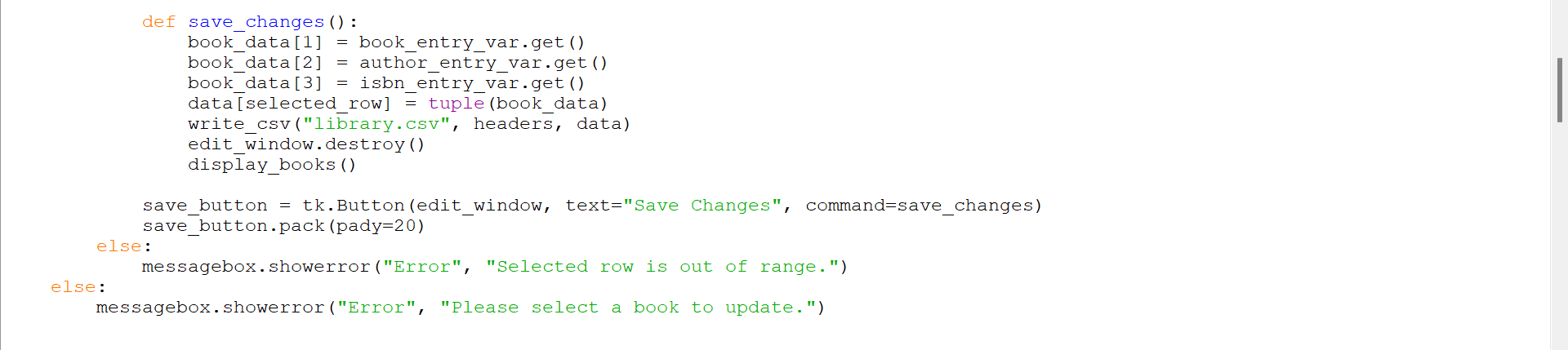


search books function

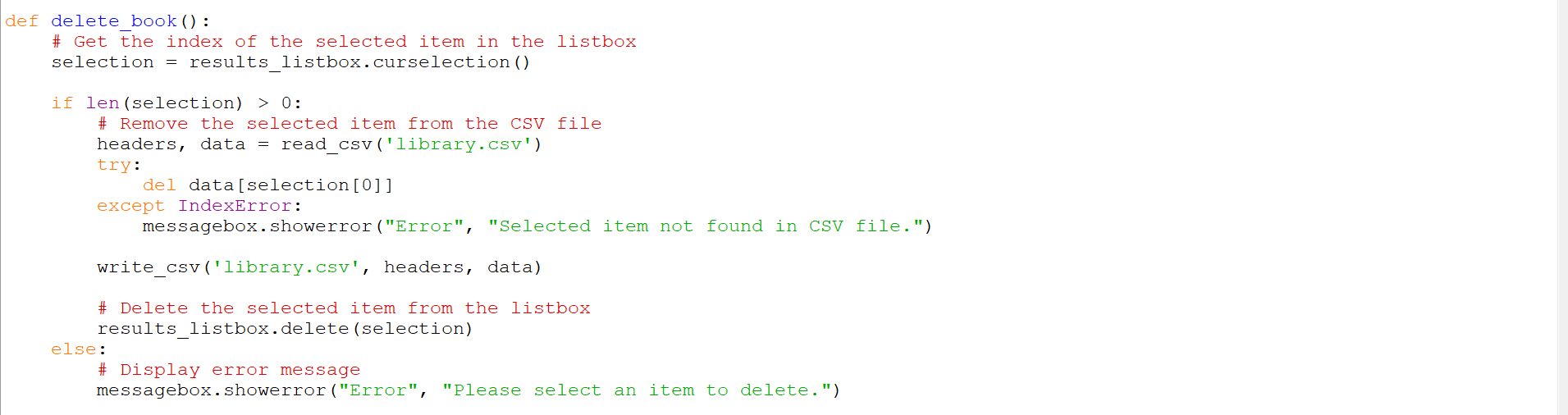


update book

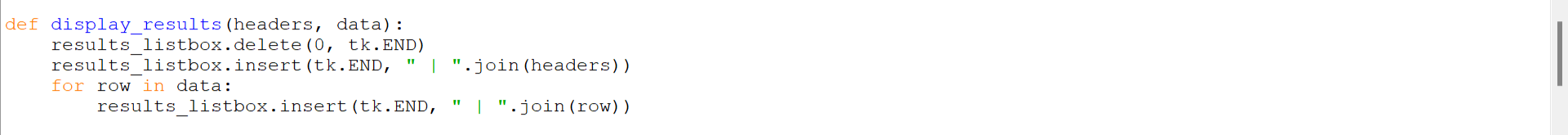




delete book function



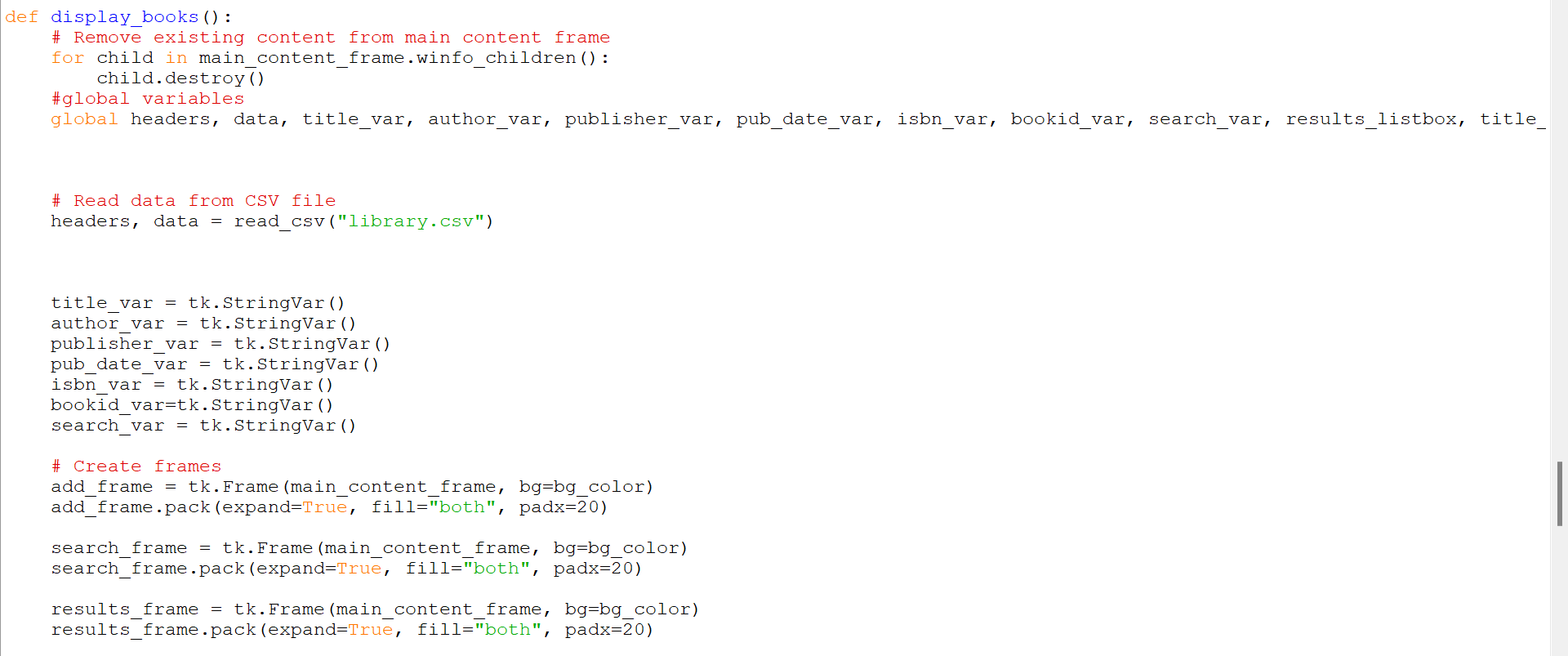
display results function

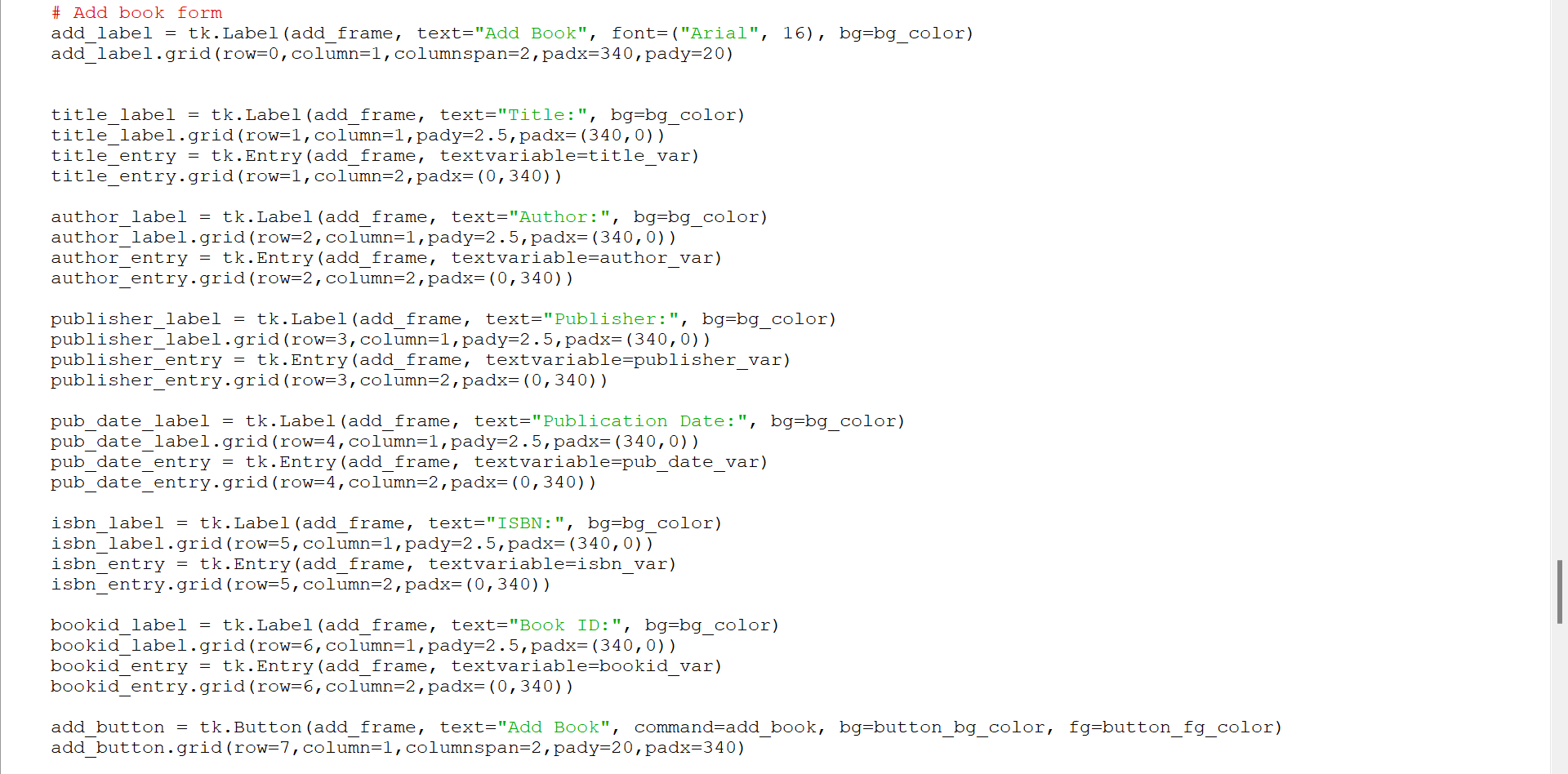


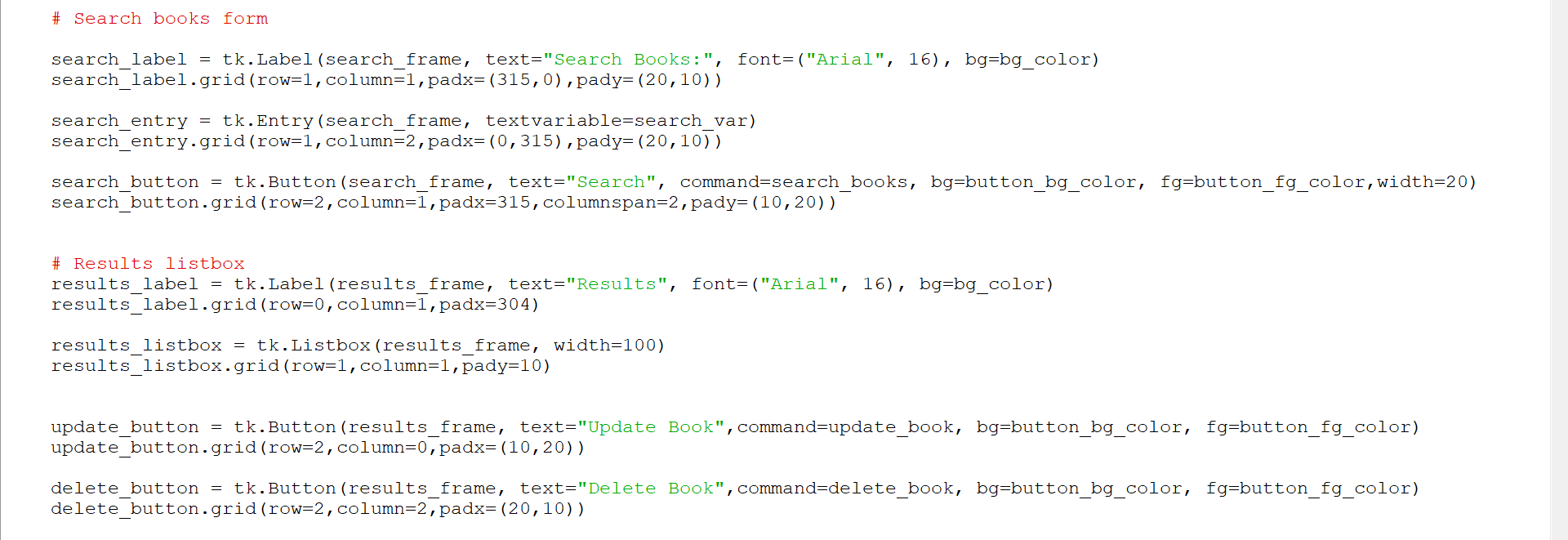
clear entries function



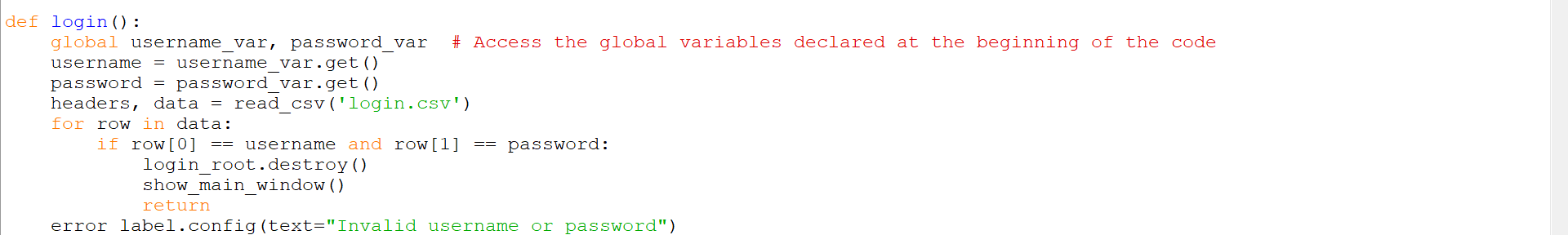
display books function



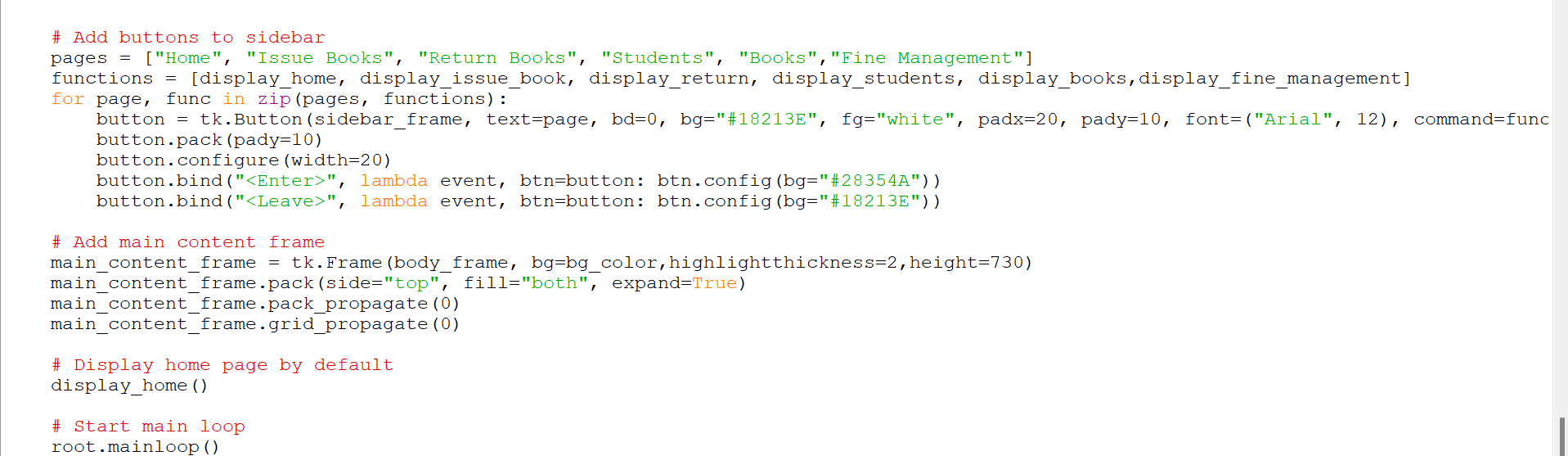
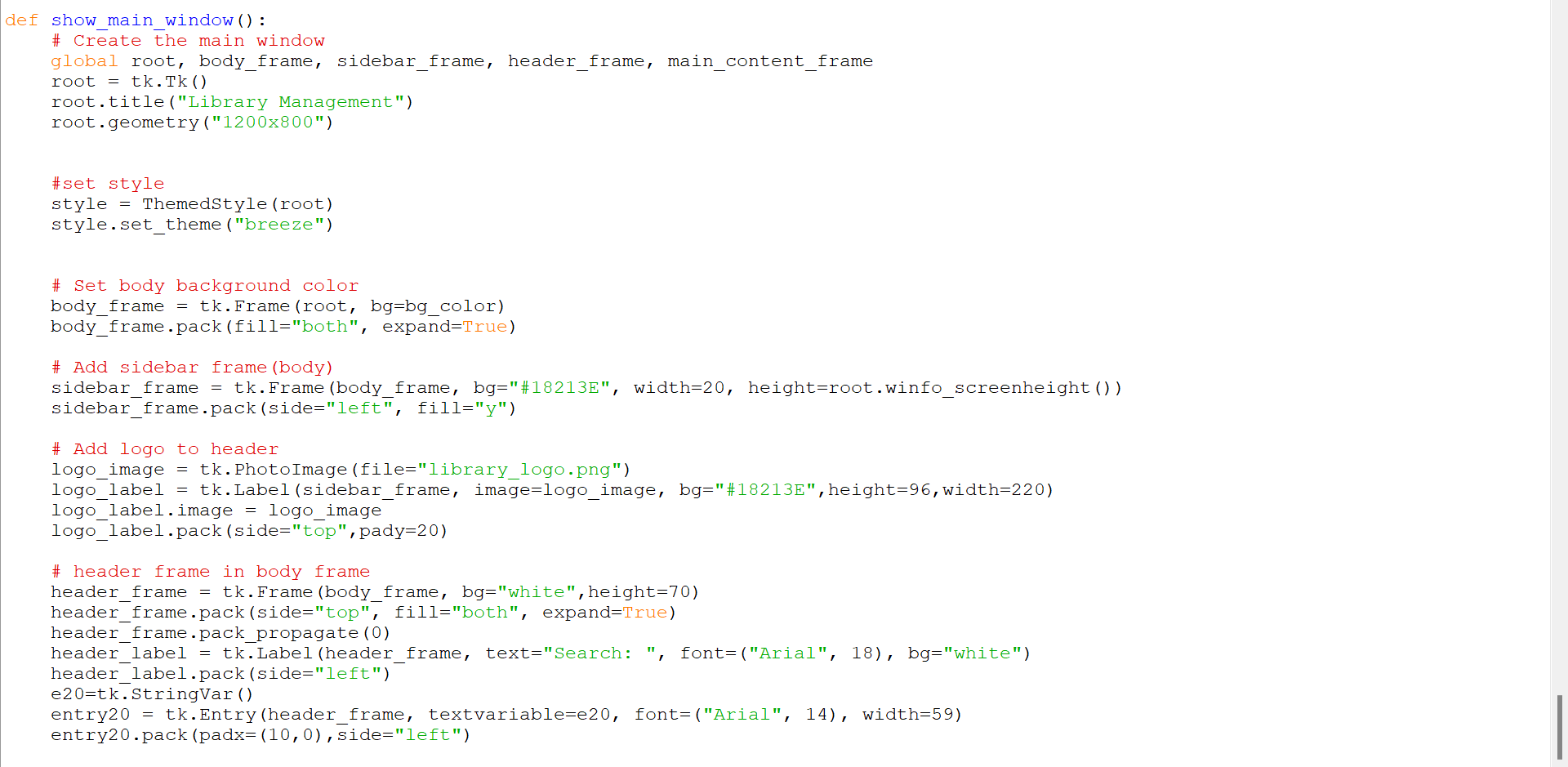




Login function



show Login frame function

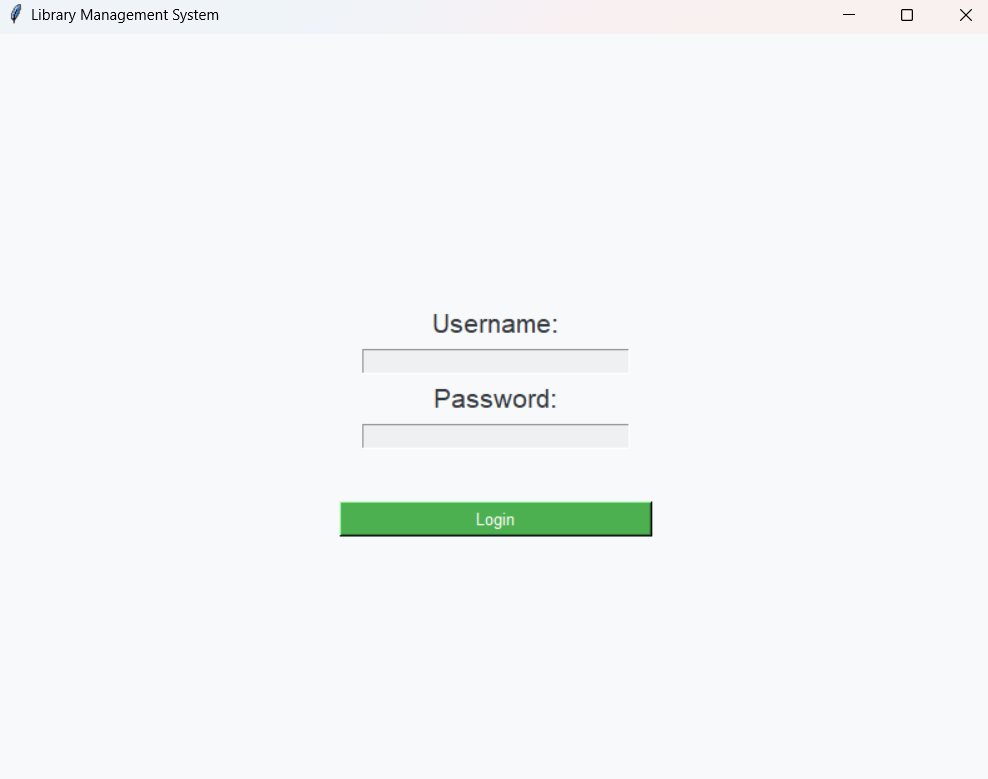


calling show login page function

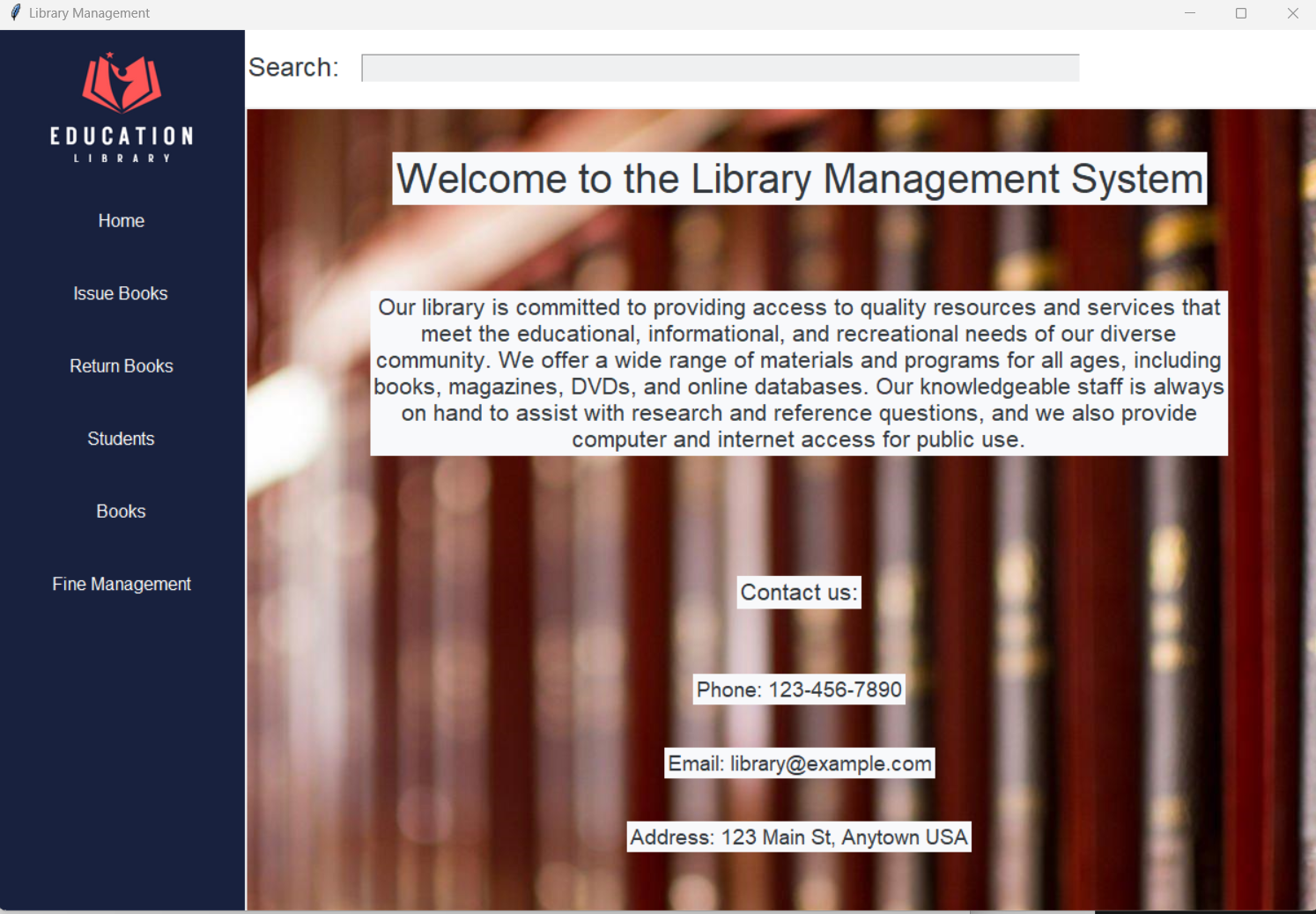


Software Screenshots:

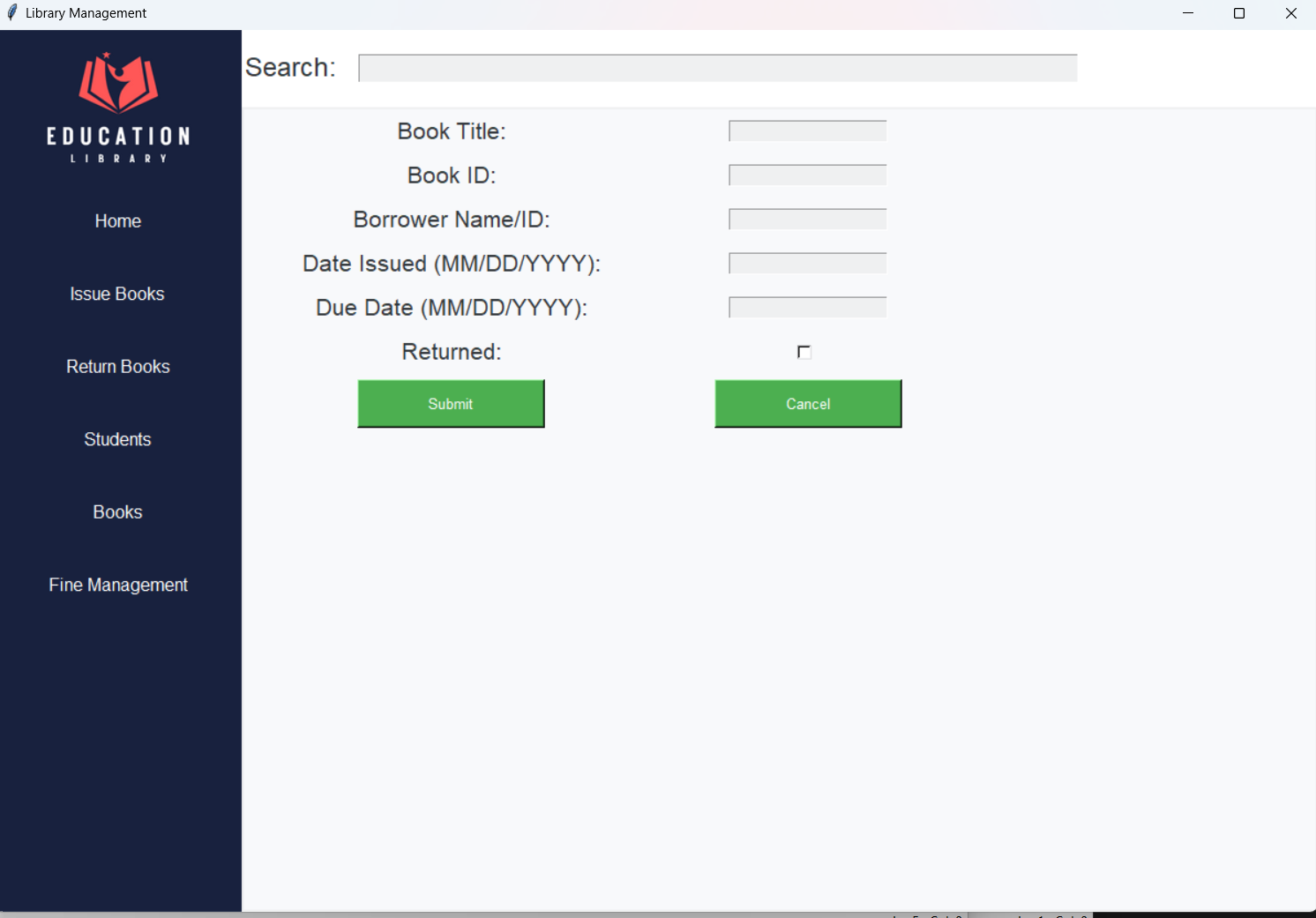
Login Page



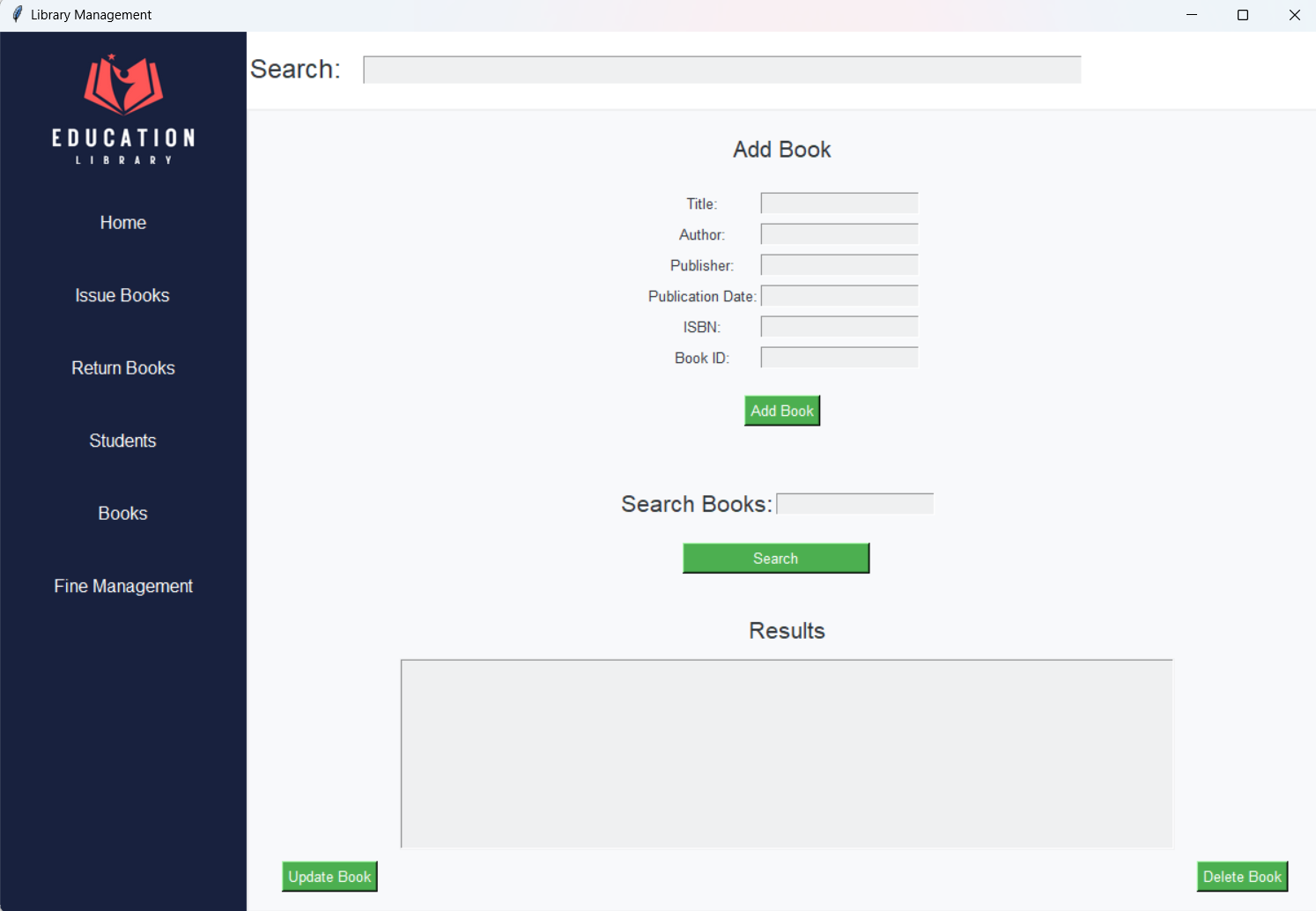
Home Page



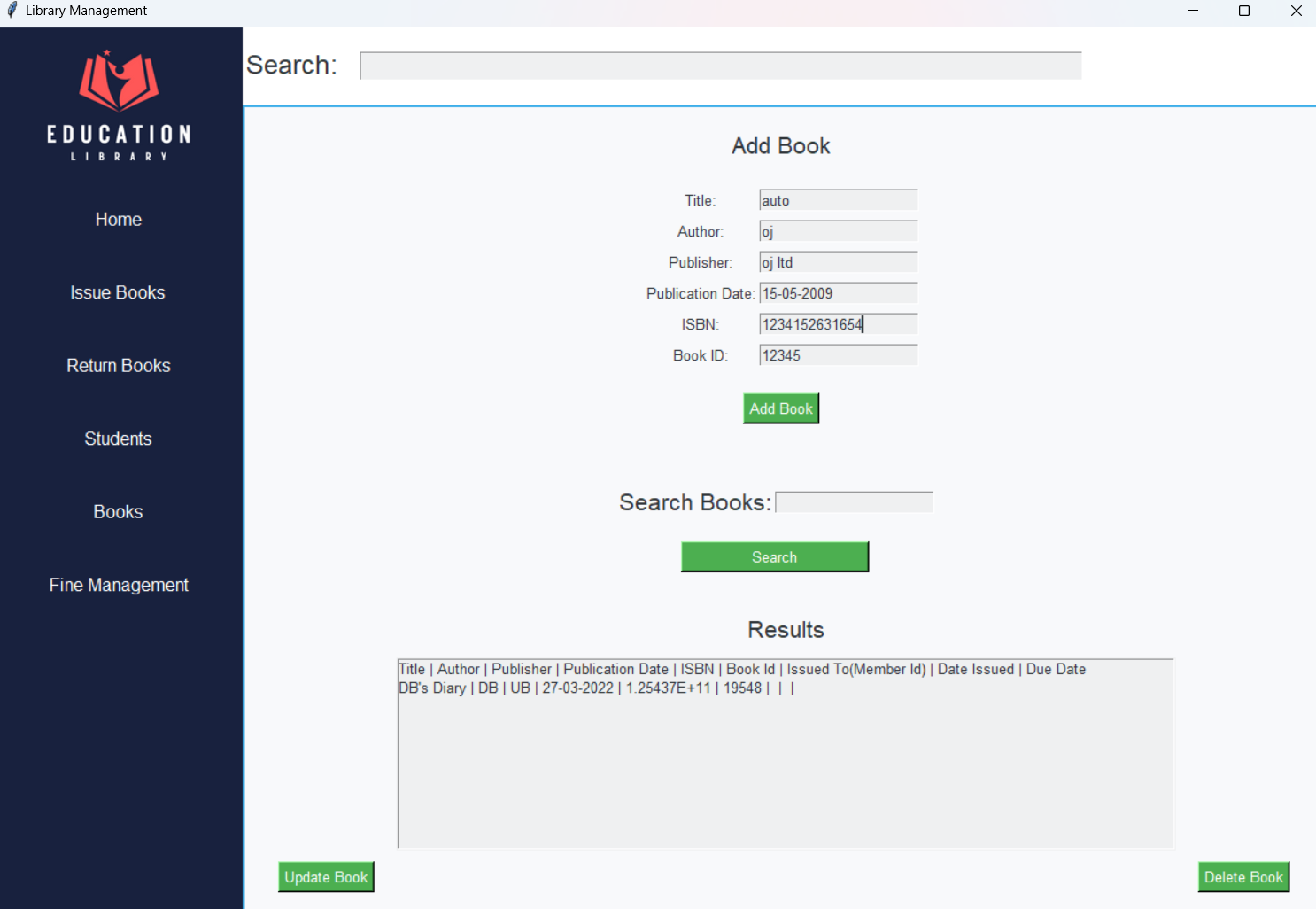
Issue Books Page



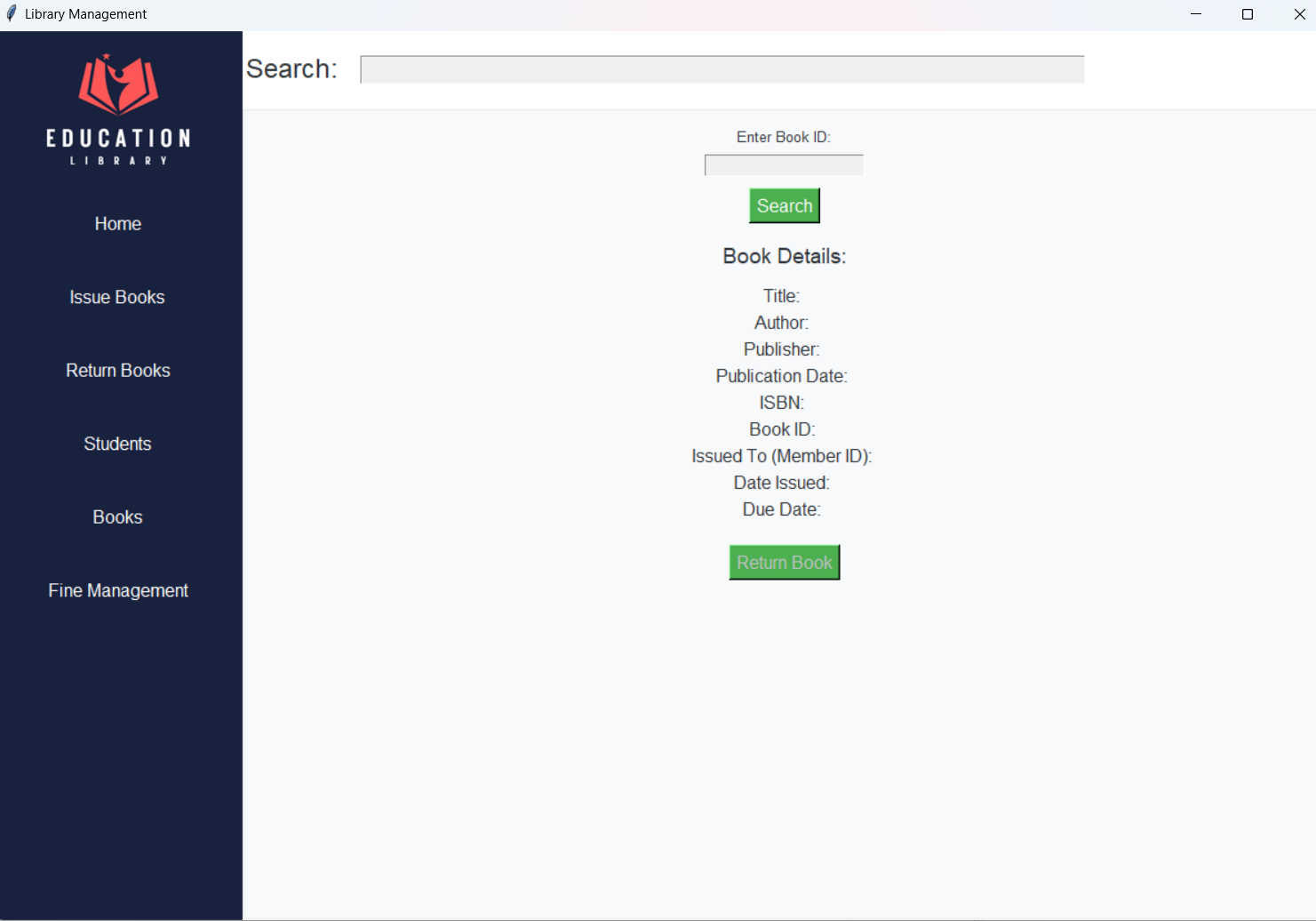
Books Tab- Empty



Books Tab – Filled



Return Books Tab



**THANK YOU**