KING MONGKUT'S INSTITUTE OF TECHNOLOGY

LATKRABANG SCHOOL OF ENGINEERING

COMPUTER ENGINEERING
DEPARTMENT

PYTHON PROJECT REPORT

INSTRUCTED & ADVISED BY: DR.PARINYA EKPARINYA

TEAM MEMBERS:

66011610 Win Thawdar Aung

66011642 May Thu Kyaw

66011666 Mihini Methasa

66011197 Rattanan Suksen

DATE OF SUB: 17/11/2023

Table of Contents

Table of Contents 2
List of Figures 5
ACKNOWLEDGEMENT 7
ABSTRACT 8
Introduction 9
Call diagram 10
Characteristics of Functions 11
Function 1: Text-based menu-driven( ): 11
Function 2: load_books(filename: str): 12
Function 3: save_books_toCSV(filename,active_books): 13
Function 4: format_table(books): 14
Function 5: list_books(input_books): 15
Function 6: show_book_info( ): 16
Function 7: insert_book( ): 17
Function 8: delete_book( ): 18
Function 9: sort_books( ): 19
Helper Functions: 19
Function 9.1: clean_title( ): 19
Function 9.2: convert_to_datetime( ): 19
Function 9.3: filter_valid_dates( ): 20
Function 10: search_books( ): 21
Function 11: clean_column_headers(books=active_books): 22
Function 12: calculate_statistics(books_list): 23
Function 13: update_books( ): 24
Test Cases 25
Function 1: Text-based menu-driven( ): 25
Function 2: load_books(filename: str) 26
Test Case 1: Valid File 26
Test Case 2: Non-Existent File 26
Function 3: save_books_toCSV(filename, active books): 27
Function 4: list_books(input_books): 27
Function 5: show_book_info( ): 28
Test Case 1: Existing book: 28
Test Case 2: Invalid BookID: 28
Function 6: insert_book( ) 29

Test Case 1: Existing BookID: 29
Test Case 2: Valid BookID: 29
Test Case 3: Cancellation: 30
Function 7: delete_book( ) 30
Test Case 1: Invalid BookID: 30
Test Case 2: Existing BookID: 30
Test Case 3: Invalid ISBN13: 31
Test Case 4: Existing ISBN13: 31
Test Case 5: Invalid average ratings: 31
Test Case 6: Valid average ratings: 32
Test Case 7: Invalid language code: 32
Test Case 8: Valid language code: 33
Test Case 9: Invalid publication date: 33
Test Case 10: Valid publication date: 33
Test Case 11: Invalid publisher: 34
Test Case 12: Valid publisher: 34
Test Case 13: Invalid Author: 34
Test Case 14: Valid Author: 35
Test Case 15: Invalid Title: 36
Test Case 16: Valid Title: 36
Function 8: sort_books( ) 37
Test Case 1: Sort by BookID - Ascending order 37
Test Case 2: Sort by BookID - Descending order 38
Test Case 3: Sort by Title - Ascending order 39
Test Case 4: Sort by Title - Descending order 40
Test Case 5: Sort by Average Ratings - Ascending order 41
Test Case 6: Sort by Average Ratings - Descending order 42
Test Case 7: Sort by Number of pages - Ascending order 43
Test Case 8: Sort by Number of pages - Descending order 44
Test Case 9: Sort by Publication Date - Ascending order 45
Test Case 10: Sort by Publication Date - Descending order 46
Test Case 11: Sort - Invalid sort key 47
Test Case 11: Sort -Invalid order 47
Function 9: search_books( ) 48
Test Case 1: Valid Input 48
Test Case 2: Invalid Input 49
Function 10: update_books( ): 50
Function 11: calculate_statistics(books_list) 50

Conclusion 51
Responsibilities of each group member:..52

List of Figures

Fig 1. Text-based menu-driven 25
Fig 2: load_books - Test Case 1 26
Fig 3: load_books - Test Case 2 26
Fig 4: save_books_toCSV 27
Fig 5: list_books - choose ‘y’ option 27
Fig 6: list_books - choose ‘n’ option 27
Fig 7: show_book_info - Test Case 1 28
Fig 8: show_book_info - Test Case 2 28
Fig 9: insert_book- Test Case 1 29
Fig 10: insert_book- Test Case 2 29
Fig 11: insert_book- Test Case 3 30
Fig 12: delete_book- Invalid BookID 30
Fig 13: delete_book- Existing BookID 30
Fig 14: delete_book- Invalid ISBN13 31
Fig 15: delete_book- Existing ISBN13 31
Fig 16: delete_book- Invalid average rating 31
Fig 17: delete_book- Valid average rating 32
Fig 18: delete_book- Invalid language code 32
Fig 19: delete_book- Valid language code 33
Fig 20: delete_book- Invalid Date 33
Fig 21: delete_book- Valid publication date 33
Fig 22: delete_book- Invalid Publisher 34
Fig 23: delete_book- Valid publisher 34
Fig 24: delete_book- Invalid Author 34
Fig 25: delete_book- Valid Author 35
Fig 26: delete_book- Invalid Title 36
Fig 27: delete_book- Valid Title 36
Fig 28: Sort by BookID - Asc order 37
Fig 29: Sort by BookID - Asc order - After calling list_books( ) 37
Fig 30: Sort by BookID - Desc order 38
Fig 31: Sort by BookID - Desc order - After calling list_books( ) 38
Fig 32: Sort by Title - Asc order 39
Fig 33: Sort by Title - Asc order - After calling list_books( ) 39
Fig 34: Sort by Title - Desc order 40
Fig 35: Sort by Title - Desc order - After calling list_books( ) 40

Fig 36: Sort by Average Ratings - Asc order..41
Fig 37: Sort by Average Ratings - Asc order - After calling list_books( )....................................41
Fig 38: Sort by Average Ratings - Desc order..42
Fig 39: Sort by Average Ratings - Desc order - After calling list_books( )..................................42
Fig 40: Sort by Number of pages - Asc order..43
Fig 41: Sort by Number of pages - Asc order - After calling list_books( )...................................43
Fig 42: Sort by Number of pages- Desc order..44
Fig 43: Sort by Number of pages - Desc order - After calling list_books( )..................................44
Fig 44: Sort by Publication Date - Asc order..45
Fig 45: Sort by Publication Date - Asc order - After calling list_books( )....................................45
Fig 46: Sort by Publication Date - Desc order..46
Fig 47: Sort by Publication Date - Desc order - After calling list_books( )..................................46
Fig 48: Sort - Invalid sort key...47
Fig 49: Sort - Invalid order...47
Fig 50: search_books - Valid Input...48
Fig 51: search_books - Invalid Input...49
Fig 52: update_books - Test Case...50
Fig 53: calculate_statistics - Test Case..50

ACKNOWLEDGEMENT

Our heartfelt gratitude goes out to everyone who helped make our project a success. This project would not have been achieved without the assistance, commitment, and knowledge of our Dr.Parinya Ekparinya and institutions. We want to start by expressing our gratitude for all of his help and advice with this project. We sincerely appreciate his contributions to our academic path, as his expertise and support were important in defining our work. We have had a much better project experience thanks to his desire to help, share knowledge, and patiently respond to inquiries.

ABSTRACT

A complete tool for managing, arranging, and accessing data about a library of books is the Book Management System, a Python-based program. The system stores book data using CSV files and provides features like loading books from files, saving the current book collection, tabulating all books, displaying specific book information, adding new books, deleting books according to different criteria, sorting books, searching books by author or title, and calculating statistical data about the collection.

A clear and maintainable code base is created by the program's modular structure, which includes functions for every particular activity. Enforcing the use of only one global variable and restricting the use of external libraries, the system makes use of fundamental Python libraries such as datetime and CSV. Through an interactive text menu that leads users through all of the features, the program encourages user participation.

Users may effectively manage their book collections, handle data modification, and find out about the statistical features of their libraries with the help of this book management system. It is a useful tool which lets users access, organize, and analyze their book data with simplicity.

Introduction

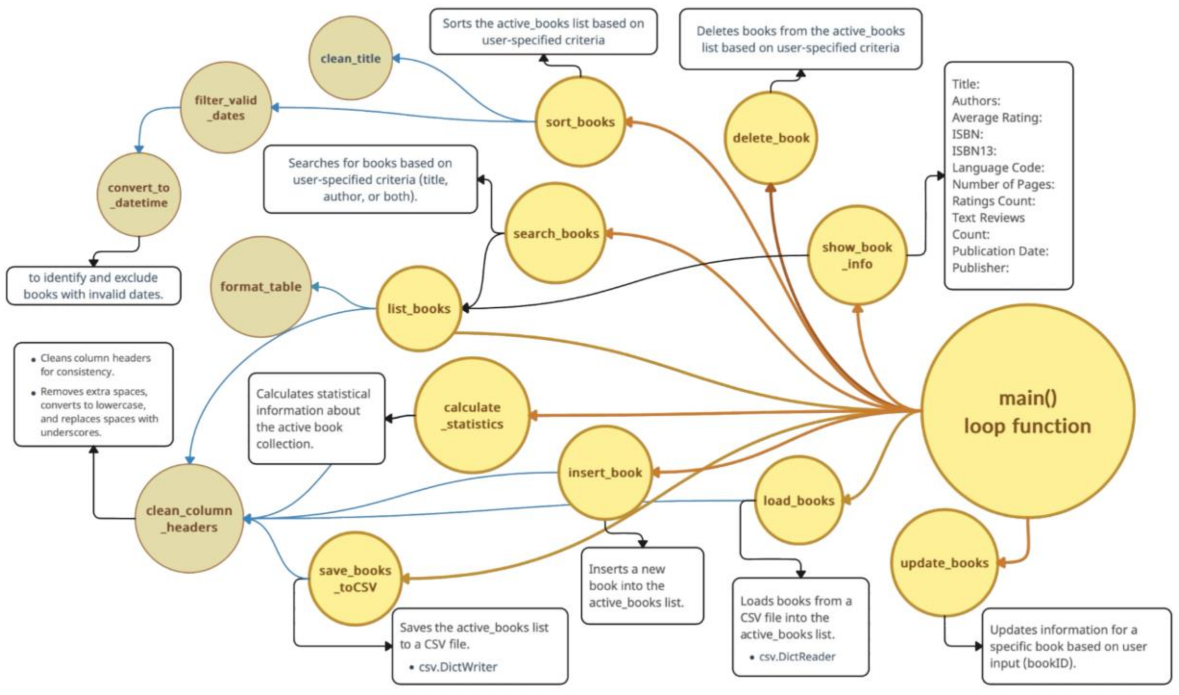
The Python code that is provided builds up a menu-driven text program that can be used to manage a library of books that are kept in a CSV file. With a user-friendly interface, this program's many capabilities enable users to engage with the book collection. The main functions involve loading books from a CSV file, listing every book, removing books according to predetermined standards, organizing books, looking up books by title or author, adding new books, importing the current book collection into a CSV file, and getting statistical data about the collection.

A global variable and a number of functions, each with a distinct function in the program's overall operation, are used to maintain the current book collection. Users may navigate and conduct activities on the book collection with ease according to the menu-driven user interface. The program also complies with a number of limitations, including using just the restricted libraries (CSV, datetime, and type), utilizing a single global variable, and following the coding and design guidelines specified in the work specifications.

The code demonstrates modularity by dividing functionalities into separate functions, each focused on a specific aspect of book management. This modular approach enhances code readability, maintainability, and allows for easy extension of functionalities. User input is actively used throughout the program to customize operations based on user preferences. Some sections use input validation to make sure user-provided data is reliable.

With all factors considered, this text-based menu-driven program provides an efficient and interactive means of managing a book collection, offering essential features for data manipulation, searching, and statistical analysis.

Call diagram


Sorts the active_books list based on user-specified criteria
Deletes books from the active_books list based on user-specified criteria
clean_title
filter_valid _dates
convert_to _datetime
to identify and exclude books with invalid dates.
sort_books
delete_book
Title:
Authors:
Average Rating:
ISBN:
ISBN13:
Language Code:
Number of Pages:
Ratings Count:
Text Reviews Count:
Publication Date:
Publisher:
Searches for books based on user-specified criteria (title, author, or both).
search_books
show_book _info
format_table
list_books
Cleans column headers for consistency.
Removes extra spaces, converts to lowercase, and replaces spaces with underscores.
Calculates statistical information about the active book collection.
calculate _statistics
main() loop function
clean_column _headers
insert_book
load_books
update_books
save_books _toCSV
Inserts a new book into the active_books list.
Loads books from a CSV file into the active_books list.
csv.DictReader
Saves the active_books list to a CSV file.
csv.DictWriter
Updates information for a specific book based on user input (bookID).

Characteristics of Functions

Function 1: Text-based menu-driven( ):

The main function, which serves as a central program loop that shows the user the menu and controls the flow depending on their input, is the main focus of the program.

Input: –

Process:

- Display a menu with a range of options for the user to select.
- Performs several tasks according to input from the user.
- Depending on the user's selection, calls the matching functions:
- Open a CSV file and load the book collection.
- Make a CSV file with the book collection.
- Write a list of every book in the library.
- Describe a specific book in detail.
- A new book should be added to the library.
- Remove the books out of the collection.
- Arrange the books in the library according to category.
- Look through the collection's books.
- Present the collection's statistics data.
- Either close the application or empty the book directory.
- Continues until the user decides to stop the program.

Output: –

Function 2: load_books(filename: str):

Add books to the active_books list by loading them from a CSV file.

Input: Path to the CSV file to load is indicated by filename (str).

Process:

- Verifies the existence of the given CSV file.
- If the file is present, use DictReader to read the contents of the CSV file.
- Applies the book entries from the file to the active_books list.
- Prevents duplication according to bookID.

Output: Print out a message if a file is found or not.

Function 3: save_books_toCSV(filename,active_books):

The method save_books_toCSV was created to take an existing list of books (active_books), remove any unnecessary column headers, and save the data to a CSV file that is given as an argument. Before publishing the data to the CSV file, the function makes sure that it is correctly formatted and encoded.

Input: The name of the CSV file containing the data from the books is indicated by the filename option. A list of dictionaries is represented by the active_books argument, and each dictionary is associated with a book that has different attributes.

Process:

- Clean the column headers of the active books to ensure consistent formatting.
- The function attempts to open the specified CSV file in write mode ('w').
- It creates a csv.DictWriter object, specifying the fieldnames (column headers) based on the cleaned books.
-
- In the CSV file, write the headers.
- Write the details of each book you iterate through in the cleaned book collection to the CSV file.

Output:

- The function prints a success message that the books have been saved to the specified file if the CSV file is successfully created and the data from the books is written.
- The function shows an error message indicating that there was a problem and that the user should try again if an error occurs during the file-writing procedure.

Function 4: format_table(books):

Using a list of books as input, the format_table function displays a formatted table containing book data. After receiving a list of dictionaries, or books, that represent books, the function produces and outputs a table with predetermined headers and column widths. The formatting ensures that the table is visually organized and aligned.

Input: A list of dictionaries is represented by the books parameter, each dictionary corresponding to a book with various attributes.

Process:

- A list of headers that correspond to the columns that will be shown in the table is initialized by the function.
- Based on the length of each header, the program determines the initial column widths.
- The function formats the table rows by generating a format_string with the calculated column widths.
- The function outputs the row containing the table header along with a dashe-filled divider line that visually divides the header from the data.
- The function uses the format string and estimated column widths to print a formatted row of data for each book in the books list.

Output: The function outputs a properly structured table with rows of data, headers, and a dividing line to the console.

Function 5: list_books(input_books):

The function forms the data into a table, cleans the column headers, and takes a list of dictionaries (input_books) as input. The table is then printed to the console, making it simple for reading the book's contents. The function offers the ability to print the data in batches if the book collection is enormous. Users can control how big datasets are displayed by using the option to print data in batches for larger collections.

Input: A list of dictionaries, each corresponding to a book that has various attributes, is represented by the input_books parameter.

Process:

- The function calls the clean_column_headers function to ensure that the column headers are cleaned and formatted consistently.
- The function initializes a list of headers that correspond to the table's columns that will be displayed.
- The length of each header is used to determine the initial column widths.
- The function formats the table rows by generating a format_string with the calculated column widths.
- The function prints the table in batches if the number of books is more than a predetermined batch size, which is set to 10.
- It uses the format_table function for each batch in order to print a prepared table.
- The function prints a single formatted table by calling the format_table function if there aren't a lot of books.

Output:

- The function outputs rows of data, headers, and a divider line to the console along with one or more well-formatted tables.
- If the book collection is empty, the function prints a message indicating that the collection is empty.
- The function messages the user that it will print the information in batches and asks input from them in order to continue printing if the collection is large

Function 6: show_book_info( ):

The purpose of the show book info feature is to provide comprehensive details about a particular book by using the bookID or ISBN13 that is entered. It searches the active book collection repeatedly for a matching book, prints the details when one is located, and so on.

Input: The user is asked to enter the ISBN13 or book ID for which they want information about.

Process:

- Each book in the active_books list is iterated over by the function.
- The input from the user is compared against the collection's book IDs and ISBN13s.
- If a match is discovered, the book's complete details including the title, authors, ratings, and more are printed.
- The user is asked whether they want to continue or exit the function.

Output:

- The function outputs comprehensive book information if a matching book is identified.
- The user gets a message that the book with the specified ID or ISBN13 was not located if there isn't a match.
- The user is asked if they wish to stop viewing the information or use the feature again.

Function 7: insert_book( ):

Users can manually add a new book to the active book collection by using the insert_book function. The user is prompted to enter the book's bookID, title, authors, and other details. In addition to making sure the user can stop the insertion process at any moment, the function validates data to prevent duplicate entries.

Input: Details about the new book, such as bookID, title, authors, average rating, ISBN, ISBN13, language code, page count, number of ratings, number of text reviews, publishing date, and publisher, are needed to be entered by the user.

Process:

- Set up a new, empty dictionary called new_book to hold the information about the new book.
- Iterate through the designated fields, obtaining values from the user in each one.
- Examine for the user input "-1," which indicates that they want to stop the insertion functioning. If that's the case, exit the function and return.
- Verify that the bookID that submitted does not already exist in the collection of books already in place in order to prevent duplicate entries. If so, let the user know and break out of the loop.
- Add the details of the new book in the active_books list if the validation is successful.
- Once the book has been successfully added, print a success message.
- Ask the user if they would like to keep putting books in or break out of the loop.

Output: When the user is entering data, display relevant messages. These can include reminders for field values, cancellation or success messages, and signals to continue or stop inserting books.

Function 8: delete_book( ):

The delete book function's purpose is to offer a varied and approachable method for removing books from the active book collection in accordance with established criteria. This feature, which lets users remove books based on several criteria, improves the way the book is managed overall.

Input:

- The user is prompted by the function to select a filter that will be deleted.
- BookID, isbn13, average ratings, language code, publication date, publisher, author, and title are some of the filters that can be selected to determine which specific information is asked.
- The user has three options: exit the program (x), return to the main menu (0), or continue on removing (y/n).

Process:

- The function uses the selected filter and input data to iterate through the active books list.
- Books that match the given criteria are found and added to the books_to_remove list.
- The books are deleted from the active_books list if they are found.
- Noticing the user if a certain book or information cannot be located, checking for valid input ranges, and offering options for ending or continuing.

Output:

- The function generates messages that show the status and outcomes of the deletion function.
- The user is informed of how many books are removed.
- The deletion procedure is verified to be finished if books are located and removed.
- It notifies the user that the requested information wasn't found if no books meet the requirements.
- The option to continue deleting, return to the main menu, or close the program appears to the user.

Function 9: sort_books( ):

Giving users the option to sort the current book collection according to specified requirements and return an ordered list. The purpose of this function is to sort a list of books by different criteria, such as book ID, title, average rating, number of pages, or publication date and return either ascending or descending ordered list. Users may view the collection in a more customisable and structured way according to this feature, which improves the book management system's overall usability.

Input:

- Active Book List - A list of books, each represented by a dictionary with keys bookID, title, average_rating, num_pages, and publication_date.
- A sorting key - User input to determine the sorting criteria (BookID, Title, Average Ratings, Number of Pages, or Publication Date).
- A sorting order - User input to specify the order of sorting (Ascending or Descending).

Helper Functions:

Function 9.1: clean_title(title):

- This function takes a book title as the argument and strips punctuations and turns alphabetic characters to lowercase and returns a string.

Function 9.2: convert_to_datetime(book):

- This function takes a book dictionary as the argument from the active_books list and returns the datetime object or the minimum possible date if the date is invalid.

Function 9.3: filter_valid_dates(books):

- This function takes a list of books as the argument and returns books that only contain valid publication dates.

Process:

- Define a global variable **active_books** to store the list of books to be sorted.
- Define a helper function **clean_title(title)** that takes a book title as an argument and returns a lowercase string with only alphabetic characters.

- Define a helper function **convert_to_datetime(book)** that takes a book dictionary as an argument and returns a datetime object representing the book's publication date, or the minimum possible date if the date is invalid.
- Define a helper function **filter_valid_dates(books)** that takes a list of books as an argument and returns a new list with only books that have valid publication dates.
- Define the main function **sort_books( )** that takes no arguments and returns the sorted list of books.
- In the main function sort_books( ),we use a while loop to prompt the user to enter a sorting group and a sorting order, and validate the user input.
- In the main function, call the filter_valid_dates( ) function to remove books with invalid dates from the active_books list.
- In the main function, use an if-elif-else statement to sort the active_books list by the chosen criteria, using the sort method and the lambda function.
- In the main function, print a message to indicate that the books have been successfully sorted and return the active_books list.

Output:

- The sorted list of books, each represented by a dictionary with keys bookID, title, average_rating, num_pages, and publication_date. based on the user-specified sorting criteria and order.
- A message confirming the successful completion of the sorting process.
- In case of invalid sorting key input or invalid order input, a message prompting the user has input invalid values.

Function 10: search_books( ):

For finding and obtaining book information from a list of active_books is the goal of the search_books function. Users can run searches using the feature according to a number of variables, including author and title or both together. Its purpose is to help users locate certain books or collections of books that meet their search criteria.

Input:

- The choice of a search description is presented to the user.
- The program prompts the user to enter the title, author, or both for the search, depending on their preference.

Process:

- The method verifies that the user's selection matches with the given options.
- The function compares the title and author or title/author with the relevant sections in active_books if the user selects to search by specific options.
- Partial matches are taken into consideration, and the search is case-insensitive.
- Books are added to the book_to_search list if any are found that meet the search parameters.
- If no books are found, a message is printed indicating that the book is not found.
- The function calls the list_books function to display the list of books matching the search criteria.

Output:

- If books matching the search criteria are found, the list of these books is displayed to the user.
- There are messages indicating that the search was successful or not successful in finding any books.
- The user is prompted to continue searching after every search process.
- The loop ends if the user chooses to exit it or back to the main menu.

Function 11: clean_column_headers(books=active_books):

The clean_column_headers function's purpose is to accept a list of dictionaries that represent books, each of which has column headers. The headers are then cleaned by replacing spaces with underscores and eliminating leading and trailing spaces. After that, a new list of dictionaries with the cleaned column headings is returned by the function.

Input:

- Books is an optional parameter that the function accepts; by default, it uses the global variable active_books.
- A list of dictionaries is represented by this parameter; each dictionary is associated with a book, and the column headers serve as the keys.

Process:

- The function iterates through each book in the provided list
- For each book, it iterates through the columns and values.
- By removing leading and trailing spaces and substituting underscores for spaces, it makes the column headings neat.
- A new dictionary is created that has the unchanged values and cleaned column headings.
- The new_list now contains this new dictionary.

Output:

- A new list (new_list) with the dictionaries for every book is what the function returns.
- The values in each dictionary have not changed despite the column headings being cleaned.

Function 12: calculate_statistics(books_list):

The calculate_statistics function uses a list of books as input to examine and calculate several statistical indicators. In order to compute statistics like the total number of books, distinct authors, distinct publishers, distinct languages, and several other insights related to the books in the list, the function firstly takes information about authors, publishers, languages, and publishing years.

Input: A list of dictionaries (books_list) is provided to the function, and each dictionary represents a book with the following attributes: title, author, publisher, page count, average rating, number of text reviews, language, and publication date.

Process:

Data Aggregation:

- Dictionary initialization is done by the function in order to compile information about authors, publishers, languages, and years.
- It gathers relevant information by going through each book in the given list iteratively.
- The data is put together for:
- Authors and the number of books each has written.
- Publishers and the number of books released by each publisher.
- Languages and the number of books written in each of languages.
- Years as well as the number of books released annually.

Statistics Calculation:

- Based on the aggregated data, several statistics are computed, including the number of books, unique authors, unique publishers, and unique languages.
- Number of books per author, publisher, language, and year, both maximum and least.
- The total number of pages in all books, as well as the average and maximum number of pages in each book.
- The average annual number of books.
- Average rating for all books together.
- Total number of text reviews.

Output:

- For the provided list of books, the function prints comprehensive statistical data, such as counts, averages, and extremums that was calculated to the console.

Function 13: update_books( ):

In response to user input, the update_books function modifies a book's particular characteristics interactively within the active_books collection. The function lets the user select a variable (title, author, etc.) to edit after asking for the bookID of the book they want to edit. The selected attribute's current value is displayed, and the user is prompted to enter the updated value. Before updating the collection, the function makes sure that the book with the given bookID is existing.

Input:

- When the user wants to update a book, they are asked to enter the bookID.
- A list of the book's characteristics, such as its title, author, and other details, is shown to the user.
- The user chooses which attribute to update.
- The updated value for the chosen attribute is provided by the user.

Process:

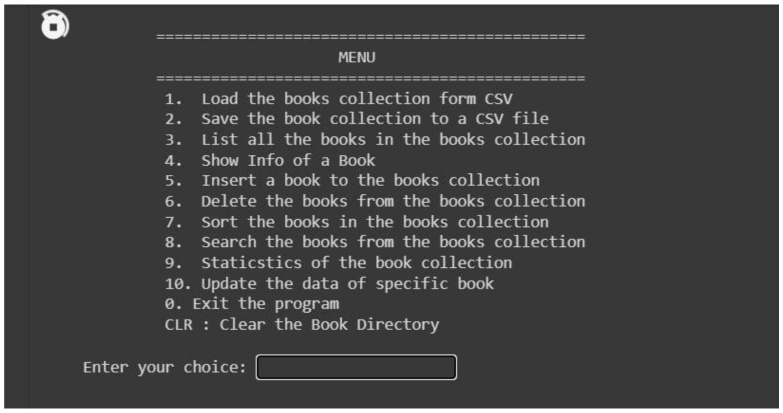
- The function checks if the active_books collection is not empty. If the collection is empty, a message is displayed, and the function does not proceed with the update.
- The function presents the user with a list of attributes from the first book in active_books.
- By entering the matching number, the user chooses an attribute.
- The function confirms the user's selection and prompts for the updated value.
- Within the collection of active books, the function locates the book with the given bookID.
- The value entered by the user is updated in the chosen attribute.

Output:

- A success message is printed by the function if the book update is successful.
- An error message appears if the given bookID cannot be located in the collection.
- The updated data for the modified book is displayed by the function.

Test Cases

Function 1: Text-based menu-driven( ):



```
    ===============================================
                      MENU
    ===============================================
     1.  Load the books collection form CSV
     2.  Save the book collection to a CSV file
     3.  List all the books in the books collection
     4.  Show Info of a Book
     5.  Insert a book to the books collection
     6.  Delete the books from the books collection
     7.  Sort the books in the books collection
     8.  Search the books from the books collection
     9.  Staticstics of the book collection
     10. Update the data of specific book
     0. Exit the program
     CLR : Clear the Book Directory

Enter your choice:
```

Fig 1. Text-based menu-driven

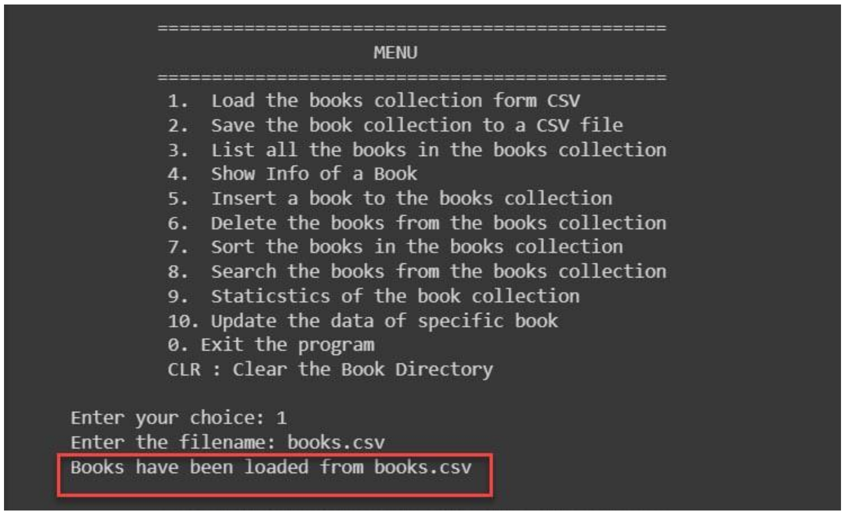
Function 2: load_books(filename: str)

Test Case 1: Valid File

Input: “books.csv”

Expected Output: Load valid CSV file and show success message.

Actual Output:



```
              ===============================================
                                   MENU
              ===============================================
               1.  Load the books collection form CSV
               2.  Save the book collection to a CSV file
               3.  List all the books in the books collection
               4.  Show Info of a Book
               5.  Insert a book to the books collection
               6.  Delete the books from the books collection
               7.  Sort the books in the books collection
               8.  Search the books from the books collection
               9.  Staticstics of the book collection
               10. Update the data of specific book
               0. Exit the program
               CLR : Clear the Book Directory

      Enter your choice: 1
      Enter the filename: books.csv
      Books have been loaded from books.csv
```

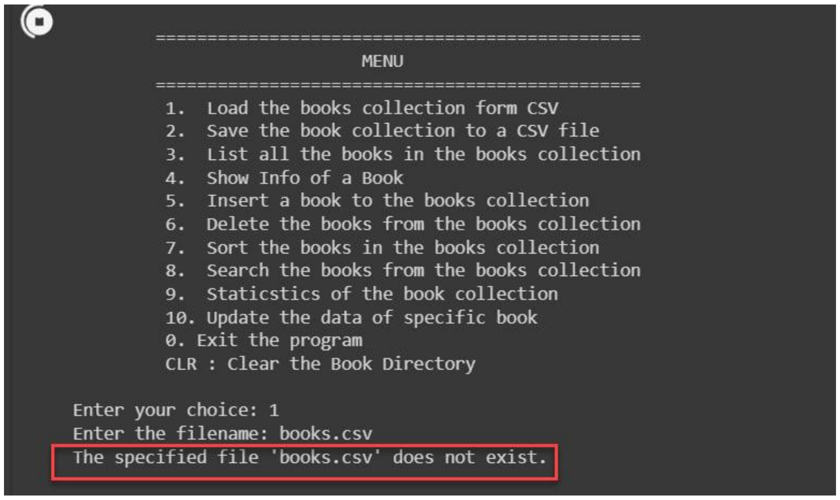
Fig 2: load_books - Test Case 1

Test Case 2: Non-Existent File

Input: “nonexistent_file.csv”

Expected Output: Error message shows when file does not exist.

Actual Output:



```
              ===============================================
                                   MENU
              ===============================================
               1.  Load the books collection form CSV
               2.  Save the book collection to a CSV file
               3.  List all the books in the books collection
               4.  Show Info of a Book
               5.  Insert a book to the books collection
               6.  Delete the books from the books collection
               7.  Sort the books in the books collection
               8.  Search the books from the books collection
               9.  Staticstics of the book collection
               10. Update the data of specific book
               0. Exit the program
               CLR : Clear the Book Directory

      Enter your choice: 1
      Enter the filename: books.csv
      The specified file 'books.csv' does not exist.
```

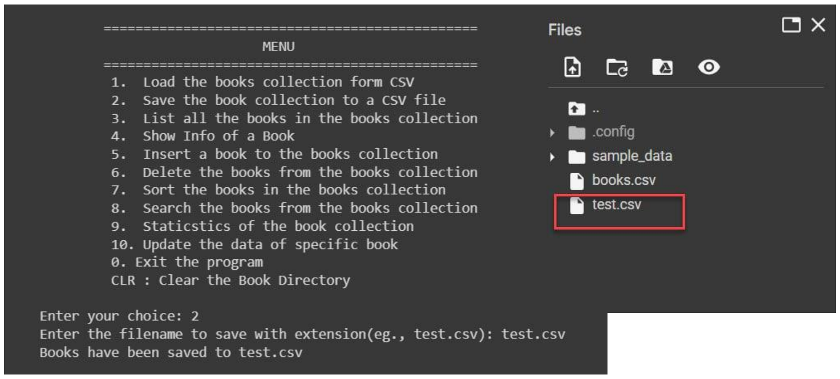
Fig 3: load_books - Test Case 2

Function 3: save_books_toCSV(filename, active books):

Input: save the new file from the user.

Expected Output: Return the successful message.

Actual Output:



```
            ==============================================
                               MENU
            ==============================================
             1.  Load the books collection form CSV
             2.  Save the book collection to a CSV file
             3.  List all the books in the books collection
             4.  Show Info of a Book
             5.  Insert a book to the books collection
             6.  Delete the books from the books collection
             7.  Sort the books in the books collection
             8.  Search the books from the books collection
             9.  Staticstics of the book collection
             10. Update the data of specific book
             0. Exit the program
             CLR : Clear the Book Directory

    Enter your choice: 2
    Enter the filename to save with extension(eg., test.csv): test.csv
    Books have been saved to test.csv
```

Fig 4: save_books_toCSV

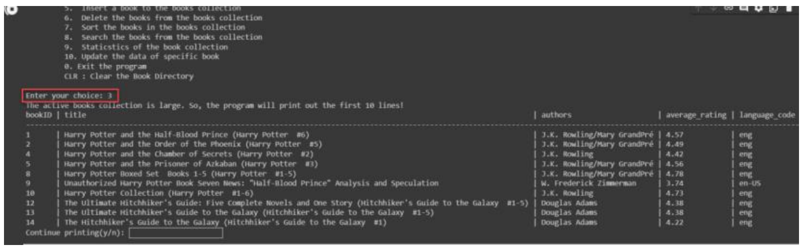
Function 4: list_books(input_books):

Input: receive the choice from the user.

Expected Output: well-formatted tables.

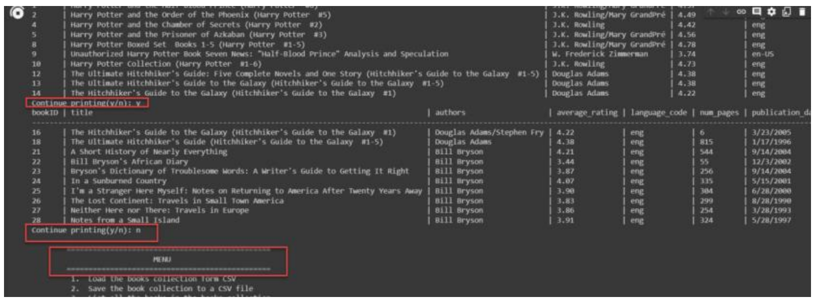
Messages will be notified for continuing or not because of the large book collection. If not, back to the menu.

Actual Output:



```
         6.  Delete the books from the books collection
         7.  Sort the books in the books collection
         8.  Search the books from the books collection
         9.  Staticstics of the book collection
         10. Update the data of specific book
         0. Exit the program
         CLR : Clear the Book Directory

  Enter your choice: 3
  The active books collection is large. So, the program will print out the first 10 lines!
  bookID | title                                                                                        | authors                   | average_rating | language_code
  ---------------------------------------------------------------------------------------------------------------------------------------------------------------------------
  1      | Harry Potter and the Half-Blood Prince (Harry Potter  #6)                                    | J.K. Rowling/Mary GrandPré | 4.57           | eng
  2      | Harry Potter and the Order of the Phoenix (Harry Potter  #5)                                 | J.K. Rowling/Mary GrandPré | 4.49           | eng
  4      | Harry Potter and the Chamber of Secrets (Harry Potter  #2)                                   | J.K. Rowling               | 4.42           | eng
  5      | Harry Potter and the Prisoner of Azkaban (Harry Potter  #3)                                  | J.K. Rowling/Mary GrandPré | 4.56           | eng
  8      | Harry Potter Boxed Set  Books 1-5 (Harry Potter  #1-5)                                       | J.K. Rowling/Mary GrandPré | 4.78           | eng
  9      | Unauthorized Harry Potter Book Seven News: "Half-Blood Prince" Analysis and Speculation      | W. Frederick Zimmerman     | 3.74           | en-US
  10     | Harry Potter Collection (Harry Potter  #1-6)                                                 | J.K. Rowling               | 4.73           | eng
  12     | The Ultimate Hitchhiker's Guide: Five Complete Novels and One Story (Hitchhiker's Guide to the Galaxy  #1-5) | Douglas Adams | 4.38   | eng
  13     | The Ultimate Hitchhiker's Guide to the Galaxy (Hitchhiker's Guide to the Galaxy  #1-5)       | Douglas Adams              | 4.38           | eng
  14     | The Hitchhiker's Guide to the Galaxy (Hitchhiker's Guide to the Galaxy  #1)                  | Douglas Adams              | 4.22           | eng
  Continue printing(y/n):
```

Fig 5: list_books - choose 'y' option



```
  2      | Harry Potter and the Order of the Phoenix (Harry Potter  #5)                                 | J.K. Rowling/Mary GrandPré | 4.49
  4      | Harry Potter and the Chamber of Secrets (Harry Potter  #2)                                   | J.K. Rowling               | 4.42           | eng
  5      | Harry Potter and the Prisoner of Azkaban (Harry Potter  #3)                                  | J.K. Rowling/Mary GrandPré | 4.56           | eng
  8      | Harry Potter Boxed Set  Books 1-5 (Harry Potter  #1-5)                                       | J.K. Rowling/Mary GrandPré | 4.78           | eng
  9      | Unauthorized Harry Potter Book Seven News: "Half-Blood Prince" Analysis and Speculation      | W. Frederick Zimmerman     | 3.74           | en-US
  10     | Harry Potter Collection (Harry Potter  #1-6)                                                 | J.K. Rowling               | 4.73           | eng
  12     | The Ultimate Hitchhiker's Guide: Five Complete Novels and One Story (Hitchhiker's Guide to the Galaxy  #1-5) | Douglas Adams | 4.38   | eng
  13     | The Ultimate Hitchhiker's Guide to the Galaxy (Hitchhiker's Guide to the Galaxy  #1-5)       | Douglas Adams              | 4.38           | eng
  14     | The Hitchhiker's Guide to the Galaxy (Hitchhiker's Guide to the Galaxy  #1)                  | Douglas Adams              | 4.22           | eng
  Continue printing(y/n): y
  bookID | title                                                                       | authors                    | average_rating | language_code | num_pages | publication_d
  ----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
  16     | The Hitchhiker's Guide to the Galaxy (Hitchhiker's Guide to the Galaxy  #1) | Douglas Adams/Stephen Fry  | 4.22           | eng           | 6         | 3/23/2005
  18     | The Ultimate Hitchhiker's Guide (Hitchhiker's Guide to the Galaxy  #1-5)    | Douglas Adams              | 4.38           | eng           | 815       | 1/17/1996
  21     | A Short History of Nearly Everything                                        | Bill Bryson                | 4.21           | eng           | 544       | 9/14/2004
  22     | Bill Bryson's African Diary                                                 | Bill Bryson                | 3.44           | eng           | 55        | 12/3/2002
  23     | Bryson's Dictionary of Troublesome Words: A Writer's Guide to Getting It Right | Bill Bryson             | 3.87           | eng           | 256       | 9/14/2004
  24     | In a Sunburned Country                                                      | Bill Bryson                | 4.07           | eng           | 335       | 5/15/2001
  25     | I'm a Stranger Here Myself: Notes on Returning to America After Twenty Years Away | Bill Bryson          | 3.90           | eng           | 304       | 6/28/2000
  26     | The Lost Continent: Travels in Small Town America                           | Bill Bryson                | 3.83           | eng           | 299       | 8/28/1990
  27     | Neither Here nor There: Travels in Europe                                   | Bill Bryson                | 3.86           | eng           | 254       | 3/28/1993
  28     | Notes from a Small Island                                                   | Bill Bryson                | 3.91           | eng           | 324       | 5/28/1997
  Continue printing(y/n): n

          ==============================================
                             MENU
          ==============================================
           1.  Load the books collection form CSV
           2.  Save the book collection to a CSV file
```

Fig 6: list_books - choose 'n' option

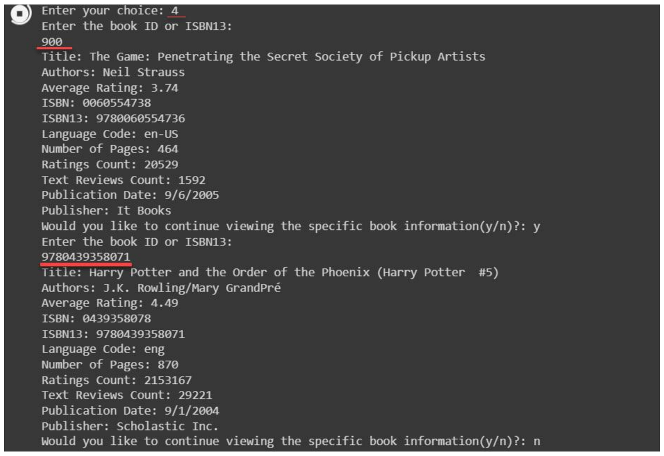
Function 5: show_book_info( ):

Test Case 1: Existing book:

Input: Existing BookID or ISBN13.

Expected Output: Print out all the data information related with input BookID or ISBN13.

Actual Output:



```
Enter your choice: 4
Enter the book ID or ISBN13:
900
Title: The Game: Penetrating the Secret Society of Pickup Artists
Authors: Neil Strauss
Average Rating: 3.74
ISBN: 0060554738
ISBN13: 9780060554736
Language Code: en-US
Number of Pages: 464
Ratings Count: 20529
Text Reviews Count: 1592
Publication Date: 9/6/2005
Publisher: It Books
Would you like to continue viewing the specific book information(y/n)?: y
Enter the book ID or ISBN13:
9780439358071
Title: Harry Potter and the Order of the Phoenix (Harry Potter  #5)
Authors: J.K. Rowling/Mary GrandPré
Average Rating: 4.49
ISBN: 0439358078
ISBN13: 9780439358071
Language Code: eng
Number of Pages: 870
Ratings Count: 2153167
Text Reviews Count: 29221
Publication Date: 9/1/2004
Publisher: Scholastic Inc.
Would you like to continue viewing the specific book information(y/n)?: n
```

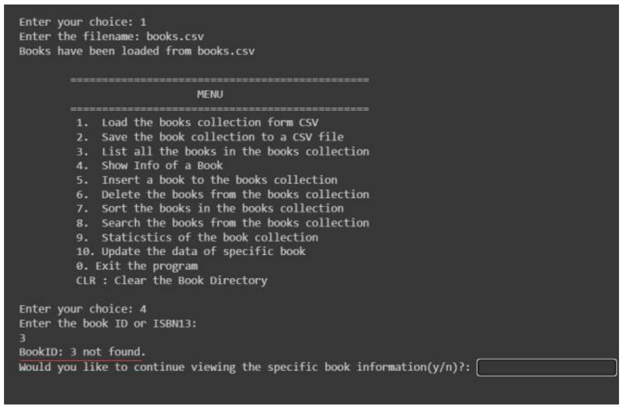
Fig 7: show_book_info - Test Case 1

Test Case 2: Invalid BookID:

Input: Nonexistent BookID or ISBN13.

Expected Output: Notify the error message ‘BookID: not found’. Print out the options to choose to continue or not.

Actual Output:



```
Enter your choice: 1
Enter the filename: books.csv
Books have been loaded from books.csv

        ===============================================
                             MENU
        ===============================================
         1.  Load the books collection form CSV
         2.  Save the book collection to a CSV file
         3.  List all the books in the books collection
         4.  Show Info of a Book
         5.  Insert a book to the books collection
         6.  Delete the books from the books collection
         7.  Sort the books in the books collection
         8.  Search the books from the books collection
         9.  Staticstics of the book collection
         10. Update the data of specific book
         0. Exit the program
         CLR : Clear the Book Directory

Enter your choice: 4
Enter the book ID or ISBN13:
3
BookID: 3 not found.
Would you like to continue viewing the specific book information(y/n)?:
```

Fig 8: show_book_info - Test Case 2

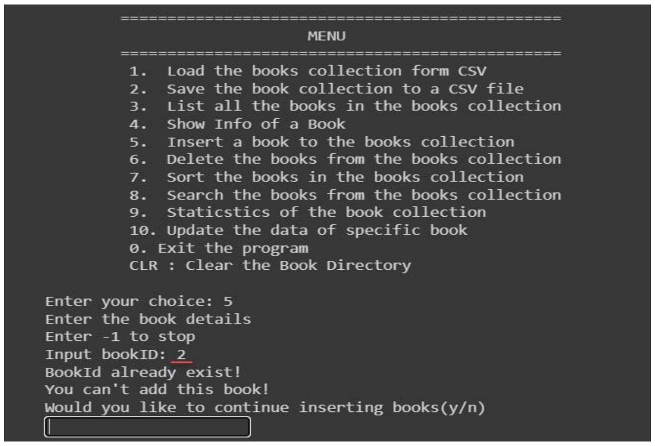
Function 6: insert_book( )

Test Case 1: Existing BookID:

Input: Enter the existing BookID

Expected Output: Show the message BookId has already existed. Ask whether you want to continue to insert or not.

Actual Output:



```
        ================================================
                             MENU
        ================================================
         1.  Load the books collection form CSV
         2.  Save the book collection to a CSV file
         3.  List all the books in the books collection
         4.  Show Info of a Book
         5.  Insert a book to the books collection
         6.  Delete the books from the books collection
         7.  Sort the books in the books collection
         8.  Search the books from the books collection
         9.  Staticstics of the book collection
         10. Update the data of specific book
         0. Exit the program
         CLR : Clear the Book Directory

Enter your choice: 5
Enter the book details
Enter -1 to stop
Input bookID: 2
BookId already exist!
You can't add this book!
Would you like to continue inserting books(y/n)
```

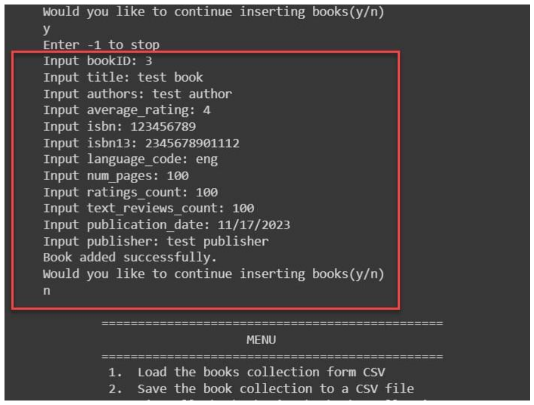
Fig 9: insert_book- Test Case 1

Test Case 2: Valid BookID:

Input: Enter new BookID

Expected Output: Display to enter detailed information for new book. After filling information, return a successful message.

Actual Output:



```
Would you like to continue inserting books(y/n)
y
Enter -1 to stop
Input bookID: 3
Input title: test book
Input authors: test author
Input average_rating: 4
Input isbn: 123456789
Input isbn13: 2345678901112
Input language_code: eng
Input num_pages: 100
Input ratings_count: 100
Input text_reviews_count: 100
Input publication_date: 11/17/2023
Input publisher: test publisher
Book added successfully.
Would you like to continue inserting books(y/n)
n
        ================================================
                             MENU
        ================================================
         1.  Load the books collection form CSV
         2.  Save the book collection to a CSV file
```

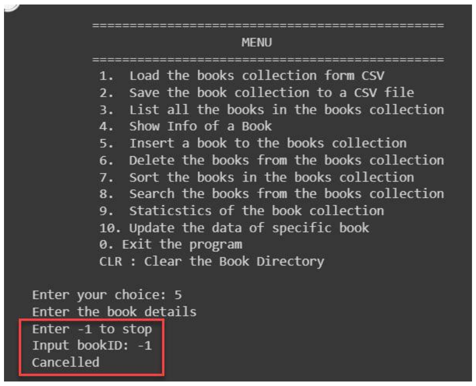
Fig 10: insert_book- Test Case 2

Test Case 3: Cancellation:

Input: Press -1 to stop.

Expected Output: Cancel message shows.

Actual Output:



```
==============================================
                     MENU
==============================================
 1.  Load the books collection form CSV
 2.  Save the book collection to a CSV file
 3.  List all the books in the books collection
 4.  Show Info of a Book
 5.  Insert a book to the books collection
 6.  Delete the books from the books collection
 7.  Sort the books in the books collection
 8.  Search the books from the books collection
 9.  Staticstics of the book collection
 10. Update the data of specific book
 0. Exit the program
 CLR : Clear the Book Directory

Enter your choice: 5
Enter the book details
Enter -1 to stop
Input bookID: -1
Cancelled
```

Fig 11: insert_book- Test Case 3

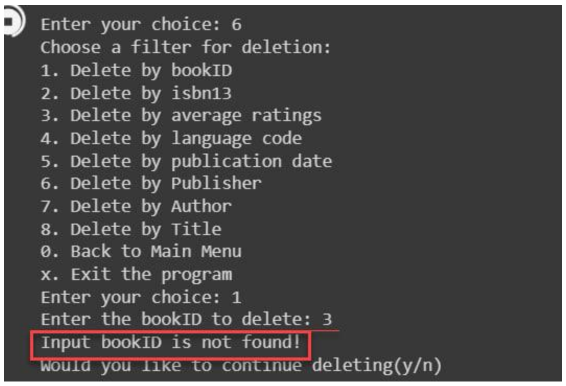
Function 7: delete_book( )

Test Case 1: Invalid BookID:

Input: User choose option 1 and enter BookID to delete.

Expected Output: Show error message if BookID is not found.

Actual Output:



```
Enter your choice: 6
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 1
Enter the bookID to delete: 3
Input bookID is not found!
Would you like to continue deleting(y/n)
```

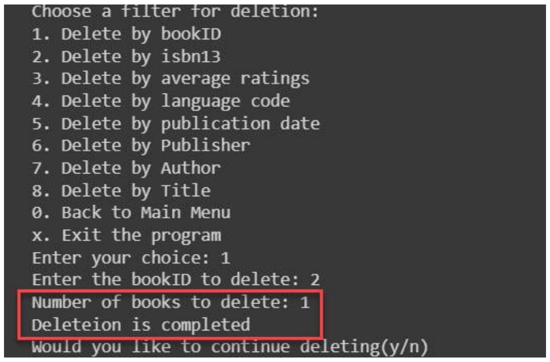
Fig 12: delete_book- Invalid BookID

Test Case 2: Existing BookID:

Input: User choose option 1 and enter BookID to delete.

Expected Output: Show the number of books which are deleted and reply completion message.

Actual Output:



```
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 1
Enter the bookID to delete: 2
Number of books to delete: 1
Deleteion is completed
Would you like to continue deleting(y/n)
```

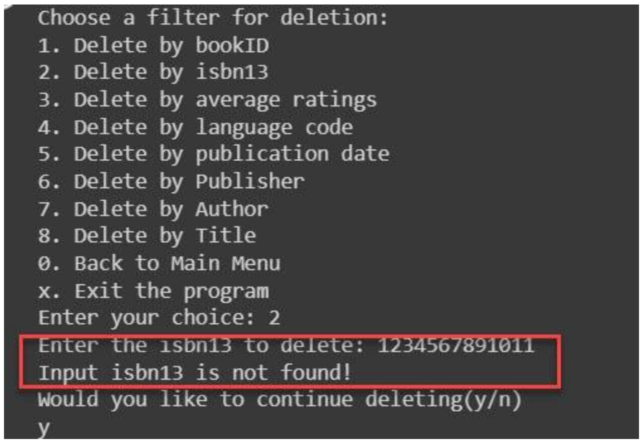
Fig 13: delete_book- Existing BookID

Test Case 3: Invalid ISBN13:

Input: User choose option 2 and enter ISBN13 to delete.

Expected Output: Show error message if ISBN13 is not found.

Actual Output:



```
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 2
Enter the isbn13 to delete: 1234567891011
Input isbn13 is not found!
Would you like to continue deleting(y/n)
y
```

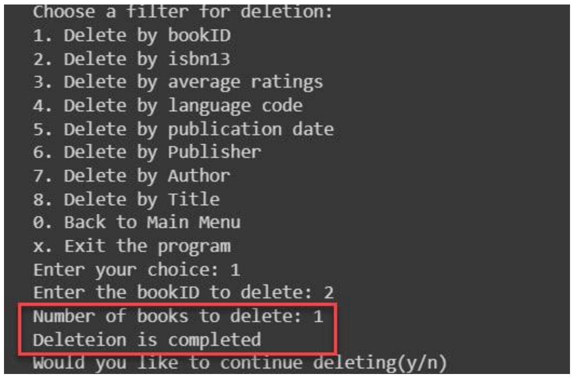
Fig 14: delete_book- Invalid ISBN13

Test Case 4: Existing ISBN13:

Input: User choose option 1 and enter ISBN13 to delete.

Expected Output: Show the number of books which are deleted and reply completion message.

Actual Output:



```
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 1
Enter the bookID to delete: 2
Number of books to delete: 1
Deleteion is completed
Would you like to continue deleting(y/n)
```

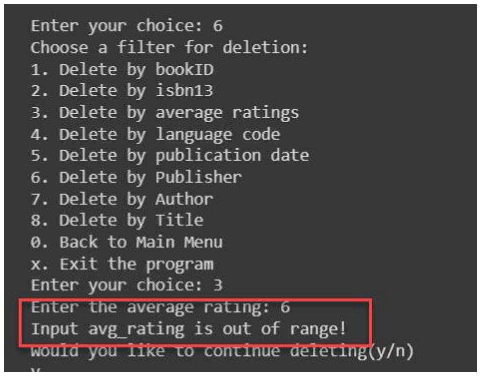
Fig 15: delete_book- Existing ISBN13

Test Case 5: Invalid average ratings:

Input: User choose option 3 and enter rating to delete.

Expected Output: Show error message if rating is out of range.

Actual Output:



```
Enter your choice: 6
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 3
Enter the average rating: 6
Input avg_rating is out of range!
Would you like to continue deleting(y/n)
```

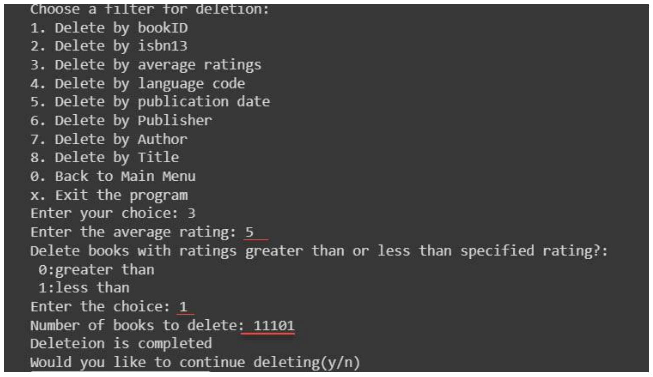
Fig 16: delete_book- Invalid average rating

Test Case 6: Valid average ratings:

Input: User choose option 3 and enter rating and 0 or 1 (greater or less) to specify to delete.

Expected Output: Show the number of books which are deleted and reply completion message. Return the message whether the user wants to continue or not.

Actual Output:



```
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 3
Enter the average rating: 5
Delete books with ratings greater than or less than specified rating?:
 0:greater than
 1:less than
Enter the choice: 1
Number of books to delete: 11101
Deleteion is completed
Would you like to continue deleting(y/n)
```

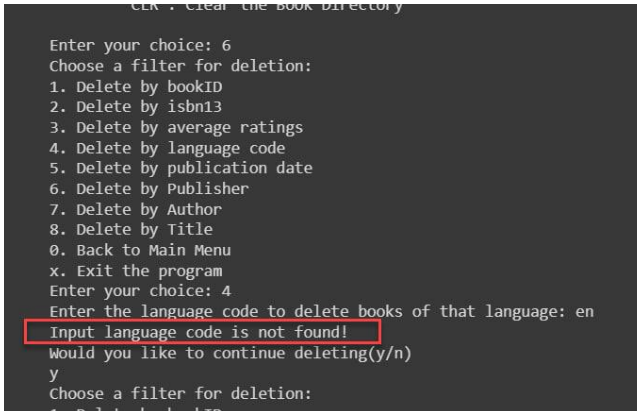
Fig 17: delete_book- Valid average rating

Test Case 7: Invalid language code:

Input: User choose option 4 and enter language code to delete.

Expected Output: Show error message if language code is not found.

Actual Output:



```
 CLR : Clear the Book Directory

Enter your choice: 6
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 4
Enter the language code to delete books of that language: en
Input language code is not found!
Would you like to continue deleting(y/n)
y
Choose a filter for deletion:
```

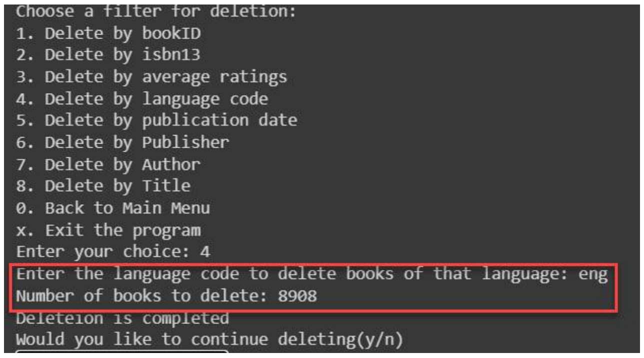
Fig 18: delete_book- Invalid language code

Test Case 8: Valid language code:

Input: User choose option 3 and enter valid language code to delete.

Expected Output: Show the number of books which are deleted and reply completion message. Return the message whether the user wants to continue or not.

Actual Output:



```
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 4
Enter the language code to delete books of that language: eng
Number of books to delete: 8908
Deleteion is completed
Would you like to continue deleting(y/n)
```

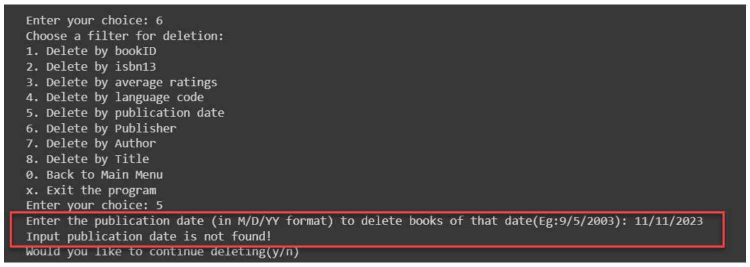
Fig 19: delete_book- Valid language code

Test Case 9: Invalid publication date:

Input: User choose option 5 and enter date to delete.

Expected Output: Show error message if the date does not exist.

Actual Output:



```
Enter your choice: 6
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 5
Enter the publication date (in M/D/YY format) to delete books of that date(Eg:9/5/2003): 11/11/2023
Input publication date is not found!
Would you like to continue deleting(y/n)
```

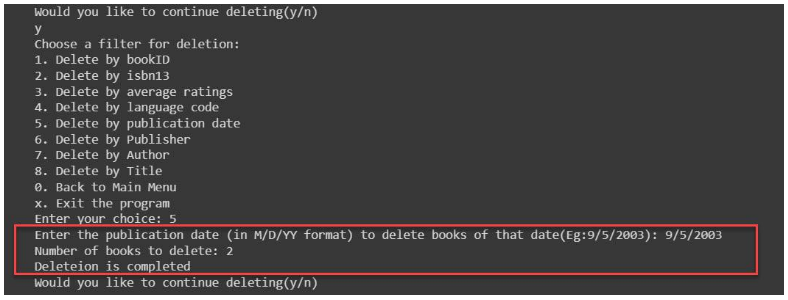
Fig 20: delete_book- Invalid Date

Test Case 10: Valid publication date:

Input: User choose option 5 and enter publication date in the correct format.

Expected Output: Show the number of books which are deleted and reply completion message. Return the message whether the user wants to continue or not.

Actual Output:



```
Would you like to continue deleting(y/n)
y
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 5
Enter the publication date (in M/D/YY format) to delete books of that date(Eg:9/5/2003): 9/5/2003
Number of books to delete: 2
Deleteion is completed
Would you like to continue deleting(y/n)
```

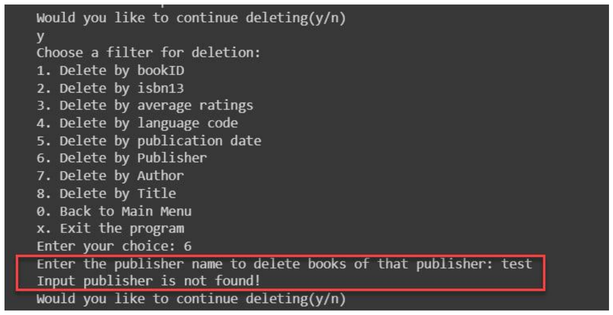
Fig 21: delete_book- Valid publication date

Test Case 11: Invalid publisher:

Input: User choose option 6 and enter name of publisher to delete.

Expected Output: Show error message if the publisher does not exist.

Actual Output:



```
Would you like to continue deleting(y/n)
y
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 6
Enter the publisher name to delete books of that publisher: test
Input publisher is not found!
Would you like to continue deleting(y/n)
```

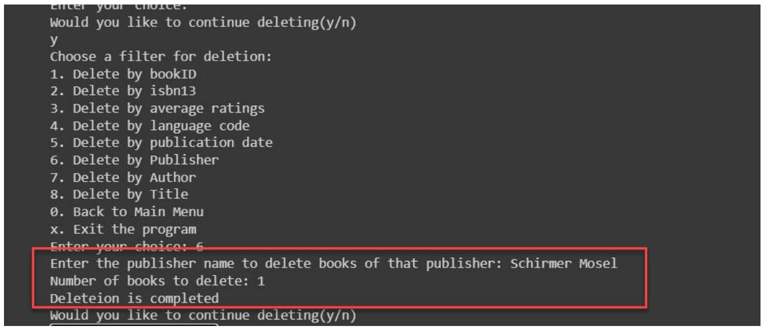
Fig 22: delete_book- Invalid Publisher

Test Case 12: Valid publisher:

Input: User chooses option 6 and enters the name of publisher.

Expected Output: Show the number of books which are deleted and reply completion message. Return the message whether the user wants to continue or not.

Actual Output:



```
Enter your choice:
Would you like to continue deleting(y/n)
y
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 6
Enter the publisher name to delete books of that publisher: Schirmer Mosel
Number of books to delete: 1
Deleteion is completed
Would you like to continue deleting(y/n)
```

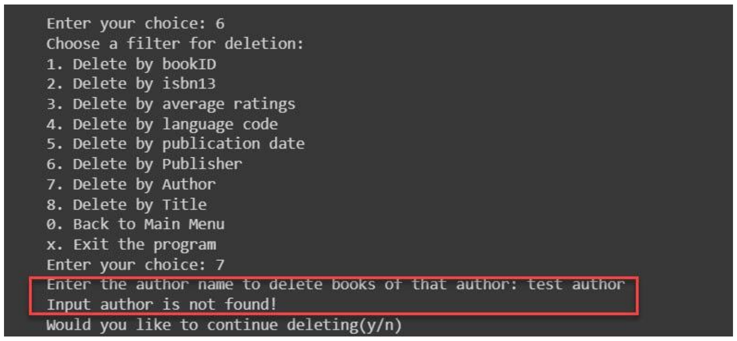
Fig 23: delete_book- Valid publisher

Test Case 13: Invalid Author:

Input: User choose option 7 and enter name of author to delete.

Expected Output: Show error message if the author does not exist.

Actual Output:



```
Enter your choice: 6
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 7
Enter the author name to delete books of that author: test author
Input author is not found!
Would you like to continue deleting(y/n)
```

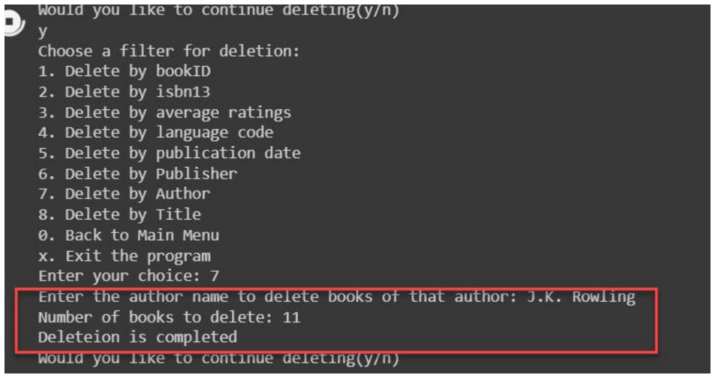
Fig 24: delete_book- Invalid Author

Test Case 14: Valid Author:

Input: User chooses option 7 and enters the name of the author.

Expected Output: Show the number of books which are deleted and reply completion message. Return the message whether the user wants to continue or not.

Actual Output:



```
Would you like to continue deleting(y/n)
y
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 7
Enter the author name to delete books of that author: J.K. Rowling
Number of books to delete: 11
Deleteion is completed
Would you like to continue deleting(y/n)
```

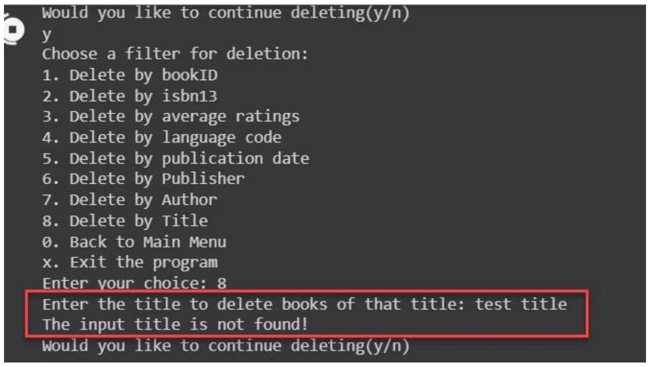
Fig 25: delete_book- Valid Author

Test Case 15: Invalid Title:

Input: User choose option 8 and enter book’s title.

Expected Output: Show error message if the title does not exist.

Actual Output:



```
Would you like to continue deleting(y/n)
y
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 8
Enter the title to delete books of that title: test title
The input title is not found!
Would you like to continue deleting(y/n)
```

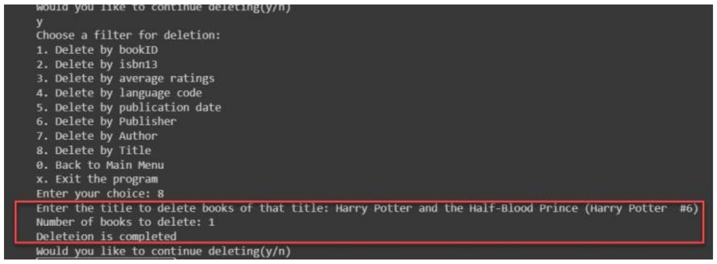
Fig 26: delete_book- Invalid Title

Test Case 16: Valid Title:

Input: User chooses option 8 and enters the book's title.

Expected Output: Show the number of books which are deleted and reply completion message. Return the message whether the user wants to continue or not.

Actual Output:



```
Would you like to continue deleting(y/n)
y
Choose a filter for deletion:
1. Delete by bookID
2. Delete by isbn13
3. Delete by average ratings
4. Delete by language code
5. Delete by publication date
6. Delete by Publisher
7. Delete by Author
8. Delete by Title
0. Back to Main Menu
x. Exit the program
Enter your choice: 8
Enter the title to delete books of that title: Harry Potter and the Half-Blood Prince (Harry Potter  #6)
Number of books to delete: 1
Deleteion is completed
Would you like to continue deleting(y/n)
```

Fig 27: delete_book- Valid Title

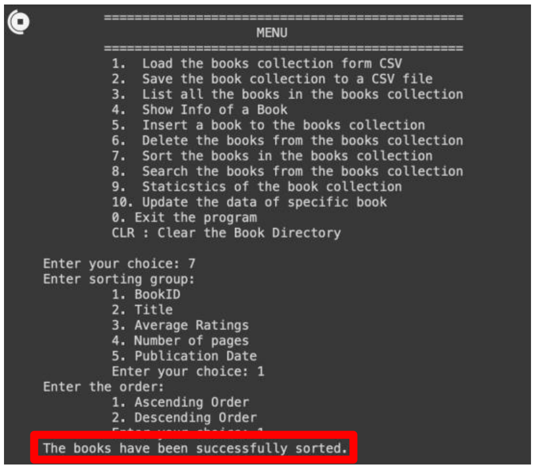
Function 8: sort_books( )

Test Case 1:   Sort by BookID - Ascending order

Input: User input number 1 for sorting group and number 1 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. “The books have been successfully sorted.” message should be prompted.

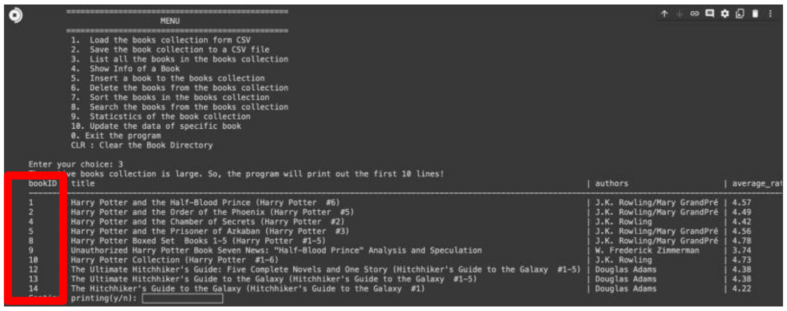
Actual output:



```
        ================================================
                              MENU
        ================================================
         1.  Load the books collection form CSV
         2.  Save the book collection to a CSV file
         3.  List all the books in the books collection
         4.  Show Info of a Book
         5.  Insert a book to the books collection
         6.  Delete the books from the books collection
         7.  Sort the books in the books collection
         8.  Search the books from the books collection
         9.  Staticstics of the book collection
         10. Update the data of specific book
         0. Exit the program
         CLR : Clear the Book Directory

Enter your choice: 7
Enter sorting group:
         1. BookID
         2. Title
         3. Average Ratings
         4. Number of pages
         5. Publication Date
         Enter your choice: 1
Enter the order:
         1. Ascending Order
         2. Descending Order
         Enter your choice: 1
The books have been successfully sorted.
```

Fig 28: Sort by BookID - Asc order

Since this function doesn’t prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
        ================================================
                              MENU
        ================================================
         1.  Load the books collection form CSV
         2.  Save the book collection to a CSV file
         3.  List all the books in the books collection
         4.  Show Info of a Book
         5.  Insert a book to the books collection
         6.  Delete the books from the books collection
         7.  Sort the books in the books collection
         8.  Search the books from the books collection
         9.  Staticstics of the book collection
         10. Update the data of specific book
         0. Exit the program
         CLR : Clear the Book Directory

Enter your choice: 3
The active books collection is large. So, the program will print out the first 10 lines!
bookID | title                                                                                   | authors                    | average_rat
----------------------------------------------------------------------------------------------------------------------------------------------
1        Harry Potter and the Half-Blood Prince (Harry Potter  #6)                                 | J.K. Rowling/Mary GrandPré | 4.57
2        Harry Potter and the Order of the Phoenix (Harry Potter  #5)                              | J.K. Rowling/Mary GrandPré | 4.49
4        Harry Potter and the Chamber of Secrets (Harry Potter  #2)                                | J.K. Rowling               | 4.42
5        Harry Potter and the Prisoner of Azkaban (Harry Potter  #3)                               | J.K. Rowling/Mary GrandPré | 4.56
8        Harry Potter Boxed Set  Books 1-5 (Harry Potter  #1-5)                                    | J.K. Rowling/Mary GrandPré | 4.78
9        Unauthorized Harry Potter Book Seven News: "Half-Blood Prince" Analysis and Speculation   | W. Frederick Zimmerman     | 3.74
10       Harry Potter Collection (Harry Potter  #1-6)                                              | J.K. Rowling               | 4.73
12       The Ultimate Hitchhiker's Guide: Five Complete Novels and One Story (Hitchhiker's Guide to the Galaxy  #1-5) | Douglas Adams | 4.38
13       The Ultimate Hitchhiker's Guide to the Galaxy (Hitchhiker's Guide to the Galaxy  #1-5)    | Douglas Adams              | 4.38
14       The Hitchhiker's Guide to the Galaxy (Hitchhiker's Guide to the Galaxy  #1)               | Douglas Adams              | 4.22
Continue printing(y/n):
```

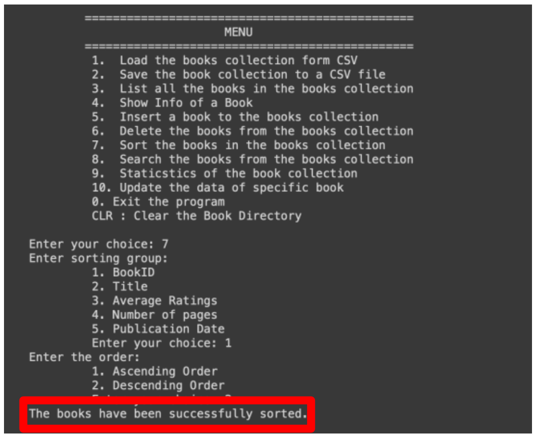
Fig 29: Sort by BookID - Asc order - After calling list_books( )

Test Case 2: Sort by BookID - Descending order

Input: User input number 1 for sorting group and number 2 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. “The books have been successfully sorted.” message should be prompted.

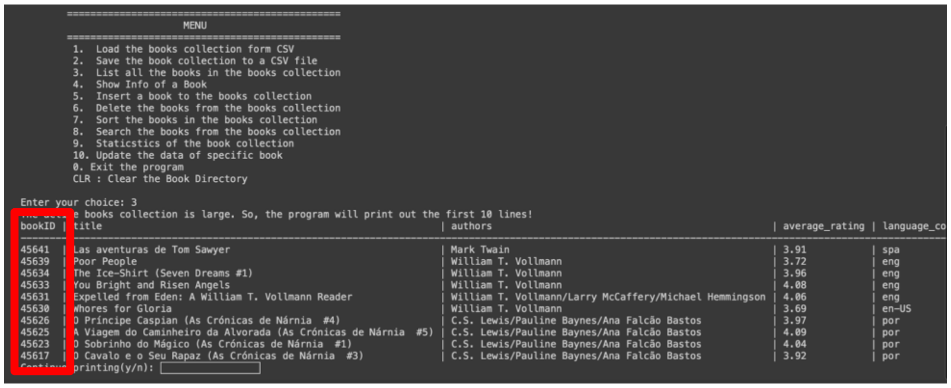
Actual output:



```
                ================================================
                                     MENU
                ================================================
                 1.  Load the books collection form CSV
                 2.  Save the book collection to a CSV file
                 3.  List all the books in the books collection
                 4.  Show Info of a Book
                 5.  Insert a book to the books collection
                 6.  Delete the books from the books collection
                 7.  Sort the books in the books collection
                 8.  Search the books from the books collection
                 9.  Staticstics of the book collection
                 10. Update the data of specific book
                 0. Exit the program
                 CLR : Clear the Book Directory

Enter your choice: 7
Enter sorting group:
         1. BookID
         2. Title
         3. Average Ratings
         4. Number of pages
         5. Publication Date
         Enter your choice: 1
Enter the order:
         1. Ascending Order
         2. Descending Order
         [illegible]
The books have been successfully sorted.
```

Fig 30: Sort by BookID - Desc order

Since this function doesn’t prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
          ================================================
                               MENU
          ================================================
           1.  Load the books collection form CSV
           2.  Save the book collection to a CSV file
           3.  List all the books in the books collection
           4.  Show Info of a Book
           5.  Insert a book to the books collection
           6.  Delete the books from the books collection
           7.  Sort the books in the books collection
           8.  Search the books from the books collection
           9.  Staticstics of the book collection
           10. Update the data of specific book
           0. Exit the program
           CLR : Clear the Book Directory

Enter your choice: 3
[illegible]e books collection is large. So, the program will print out the first 10 lines!
bookID | title                                                    | authors                                              | average_rating | language_co
---------------------------------------------------------------------------------------------------------------------------------------------------------

45641  | Las aventuras de Tom Sawyer                              | Mark Twain                                           | 3.91           | spa
45639  | Poor People                                              | William T. Vollmann                                  | 3.72           | eng
45634  | The Ice-Shirt (Seven Dreams #1)                          | William T. Vollmann                                  | 3.96           | eng
45633  | You Bright and Risen Angels                              | William T. Vollmann                                  | 4.08           | eng
45631  | Expelled from Eden: A William T. Vollmann Reader         | William T. Vollmann/Larry McCaffery/Michael Hemmingson | 4.06         | eng
45630  | Whores for Gloria                                        | William T. Vollmann                                  | 3.69           | en-US
45626  | O Príncipe Caspian (As Crónicas de Nárnia  #4)           | C.S. Lewis/Pauline Baynes/Ana Falcão Bastos          | 3.97           | por
45625  | A Viagem do Caminheiro da Alvorada (As Crónicas de Nárnia  #5) | C.S. Lewis/Pauline Baynes/Ana Falcão Bastos    | 4.09           | por
45623  | O Sobrinho do Mágico (As Crónicas de Nárnia  #1)         | C.S. Lewis/Pauline Baynes/Ana Falcão Bastos          | 4.04           | por
45617  | O Cavalo e o Seu Rapaz (As Crónicas de Nárnia  #3)       | C.S. Lewis/Pauline Baynes/Ana Falcão Bastos          | 3.92           | por
[illegible] printing(y/n):
```

Fig 31: Sort by BookID - Desc order - After calling list_books( )

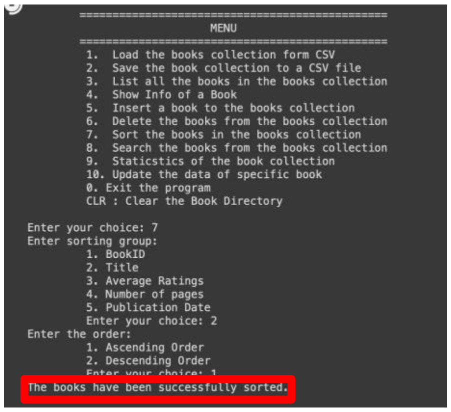
Test Case 3: Sort by Title - Ascending order

Helper Function : clean_title(title)

Input: User input number 2 for sorting group and number 1 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. “The books have been successfully sorted.” message should be prompted.

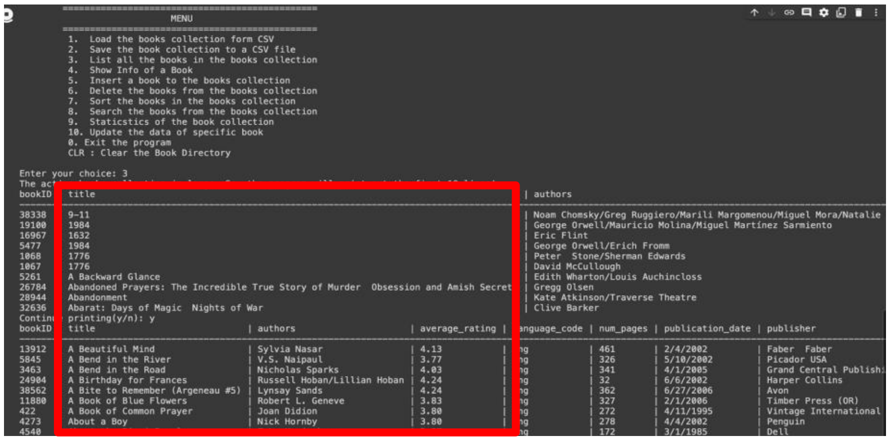
Actual output:



```
          ==============================================
                              MENU
          ==============================================
           1.  Load the books collection form CSV
           2.  Save the book collection to a CSV file
           3.  List all the books in the books collection
           4.  Show Info of a Book
           5.  Insert a book to the books collection
           6.  Delete the books from the books collection
           7.  Sort the books in the books collection
           8.  Search the books from the books collection
           9.  Staticstics of the book collection
           10. Update the data of specific book
           0. Exit the program
           CLR : Clear the Book Directory

  Enter your choice: 7
  Enter sorting group:
           1. BookID
           2. Title
           3. Average Ratings
           4. Number of pages
           5. Publication Date
           Enter your choice: 2
  Enter the order:
           1. Ascending Order
           2. Descending Order
           Enter your choice: 1
  The books have been successfully sorted.
```

Fig 32: Sort by Title - Asc order

Since this function doesn’t prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
          ==============================================
                              MENU
          ==============================================
           1.  Load the books collection form CSV
           2.  Save the book collection to a CSV file
           3.  List all the books in the books collection
           4.  Show Info of a Book
           5.  Insert a book to the books collection
           6.  Delete the books from the books collection
           7.  Sort the books in the books collection
           8.  Search the books from the books collection
           9.  Staticstics of the book collection
           10. Update the data of specific book
           0. Exit the program
           CLR : Clear the Book Directory

  Enter your choice: 3
  The act
  bookID   title                                                                          | authors
  ---------------------------------------------------------------------------------------------------------------
  38338    9-11                                                                           | Noam Chomsky/Greg Ruggiero/Marili Margomenou/Miguel Mora/Natalie
  19100    1984                                                                           | George Orwell/Mauricio Molina/Miguel Martínez Sarmiento
  16967    1632                                                                           | Eric Flint
  5477     1984                                                                           | George Orwell/Erich Fromm
  1068     1776                                                                           | Peter  Stone/Sherman Edwards
  1067     1776                                                                           | David McCullough
  5261     A Backward Glance                                                              | Edith Wharton/Louis Auchincloss
  26784    Abandoned Prayers: The Incredible True Story of Murder  Obsession and Amish Secret | Gregg Olsen
  28944    Abandonment                                                                    | Kate Atkinson/Traverse Theatre
  32636    Abarat: Days of Magic  Nights of War                                           | Clive Barker
  Contin   printing(y/n): y
  bookID   title                                | authors                       | average_rating | anguage_code | num_pages | publication_date | publisher
  ---------------------------------------------------------------------------------------------------------------
  13912    A Beautiful Mind                     | Sylvia Nasar                  | 4.13           | ng           | 461       | 2/4/2002         | Faber  Faber
  5845     A Bend in the River                  | V.S. Naipaul                  | 3.77           | ng           | 326       | 5/10/2002        | Picador USA
  3463     A Bend in the Road                   | Nicholas Sparks               | 4.03           | ng           | 341       | 4/1/2005         | Grand Central Publishi
  24904    A Birthday for Frances               | Russell Hoban/Lillian Hoban   | 4.24           | ng           | 32        | 6/6/2002         | Harper Collins
  38562    A Bite to Remember (Argeneau #5)     | Lynsay Sands                  | 4.24           | ng           | 362       | 6/27/2006        | Avon
  11880    A Book of Blue Flowers               | Robert L. Geneve              | 3.83           | ng           | 327       | 2/1/2006         | Timber Press (OR)
  422      A Book of Common Prayer              | Joan Didion                   | 3.80           | ng           | 272       | 4/11/1995        | Vintage International
  4273     About a Boy                          | Nick Hornby                   | 3.80           | ng           | 278       | 4/4/2002         | Penguin
  4540                                                                                         | ng           | 172       | 3/1/1985         | Dell
```

Fig 33: Sort by Title - Asc order - After calling list_books( )

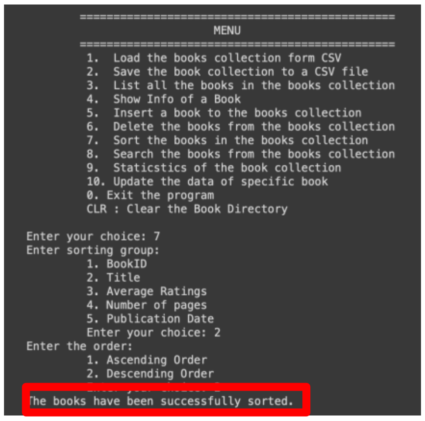
Test Case 4: Sort by Title - Descending order

Helper Function : clean_title(title)

Input: User input number 2 for sorting group and number 2 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. “The books have been successfully sorted.” message should be prompted.

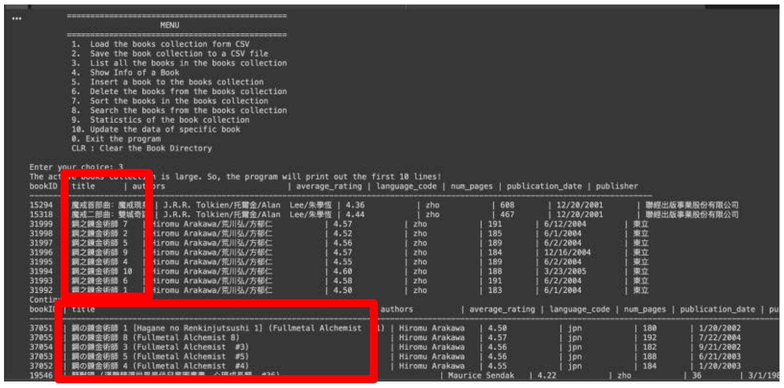
Actual output:



```
                ================================================
                                     MENU
                ================================================
                 1.  Load the books collection form CSV
                 2.  Save the book collection to a CSV file
                 3.  List all the books in the books collection
                 4.  Show Info of a Book
                 5.  Insert a book to the books collection
                 6.  Delete the books from the books collection
                 7.  Sort the books in the books collection
                 8.  Search the books from the books collection
                 9.  Staticstics of the book collection
                 10. Update the data of specific book
                 0. Exit the program
                 CLR : Clear the Book Directory

Enter your choice: 7
Enter sorting group:
         1. BookID
         2. Title
         3. Average Ratings
         4. Number of pages
         5. Publication Date
         Enter your choice: 2
Enter the order:
         1. Ascending Order
         2. Descending Order
The books have been successfully sorted.
```

Fig 34: Sort by Title - Desc order

Since this function doesn’t prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
        ================================================
                             MENU
        ================================================
         1.  Load the books collection form CSV
         2.  Save the book collection to a CSV file
         3.  List all the books in the books collection
         4.  Show Info of a Book
         5.  Insert a book to the books collection
         6.  Delete the books from the books collection
         7.  Sort the books in the books collection
         8.  Search the books from the books collection
         9.  Staticstics of the book collection
         10. Update the data of specific book
         0. Exit the program
         CLR : Clear the Book Directory

Enter your choice: 3
The act                 is large. So, the program will print out the first 10 lines!
bookID  title      | aut  rs                        | average_rating | language_code | num_pages | publication_date | publisher
---------------------------------------------------------------------------------------------------------------------------
15294   魔戒首部曲: 魔戒現 | J.R.R. Tolkien/托爾金/Alan  Lee/朱學恆 | 4.36      | zho       | 608      | 12/20/2001      | 聯經出版事業股份有限公司
15318   魔戒二部曲: 雙城奇 | J.R.R. Tolkien/托爾金/Alan  Lee/朱學恆 | 4.44      | zho       | 467      | 12/20/2001      | 聯經出版事業股份有限公司
31999   鋼之鍊金術師 7   | Hiromu Arakawa/荒川弘/方郁仁      | 4.57      | zho     | 191     | 6/12/2004     | 東立
31998   鋼之鍊金術師 2   | Hiromu Arakawa/荒川弘/方郁仁      | 4.52      | zho     | 185     | 6/1/2004      | 東立
31997   鋼之鍊金術師 5   | Hiromu Arakawa/荒川弘/方郁仁      | 4.56      | zho     | 189     | 6/2/2004      | 東立
31996   鋼之鍊金術師 9   | Hiromu Arakawa/荒川弘/方郁仁      | 4.57      | zho     | 184     | 12/16/2004    | 東立
31995   鋼之鍊金術師 4   | Hiromu Arakawa/荒川弘/方郁仁      | 4.55      | zho     | 189     | 6/2/2004      | 東立
31994   鋼之鍊金術師 10  | Hiromu Arakawa/荒川弘/方郁仁      | 4.60      | zho     | 188     | 3/23/2005     | 東立
31993   鋼之鍊金術師 6   | Hiromu Arakawa/荒川弘/方郁仁      | 4.58      | zho     | 191     | 6/2/2004      | 東立
31992   鋼之鍊金術師 1   | Hiromu Arakawa/荒川弘/方郁仁      | 4.50      | zho     | 183     | 6/1/2004      | 東立
Contin
bookI  | title                                              authors          | average_rating | language_code | num_pages | publication_date | pu
---------------------------------------------------------------------------------------------------------------------------
37051  | 鋼の錬金術師 1 [Hagane no Renkinjutsushi 1] (Fullmetal Alchemist  1) | Hiromu Arakawa  | 4.50   | jpn   | 180   | 1/20/2002
37055  | 鋼の錬金術師 8 (Fullmetal Alchemist 8)                          | Hiromu Arakawa  | 4.57   | jpn   | 192   | 7/22/2004
37054  | 鋼の錬金術師 3 (Fullmetal Alchemist  #3)                        | Hiromu Arakawa  | 4.56   | jpn   | 182   | 9/21/2002
37053  | 鋼の錬金術師 5 (Fullmetal Alchemist  #5)                        | Hiromu Arakawa  | 4.56   | jpn   | 188   | 6/21/2003
37052  | 鋼の錬金術師 4 (Fullmetal Alchemist  #4)                        | Hiromu Arakawa  | 4.55   | jpn   | 184   | 1/20/2003
19546  | [illegible]                                          | Maurice Sendak  | 4.22       | zho     | 36     | 3/1/19
```

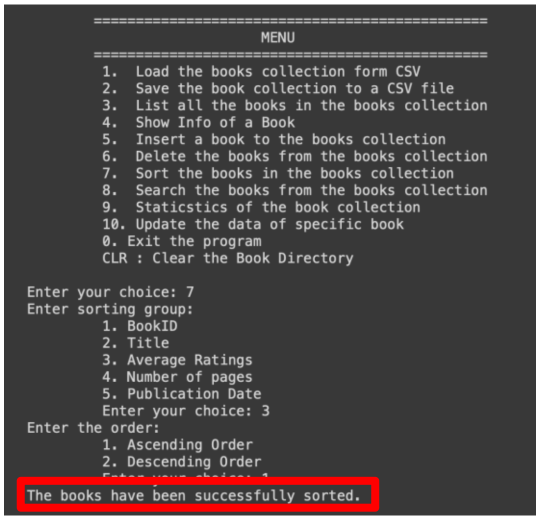
Fig 35: Sort by Title - Desc order - After calling list_books( )

Test Case 5: Sort by Average Ratings - Ascending order

Input: User input number 3 for sorting group and number 1 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. “The books have been successfully sorted.” message should be prompted.

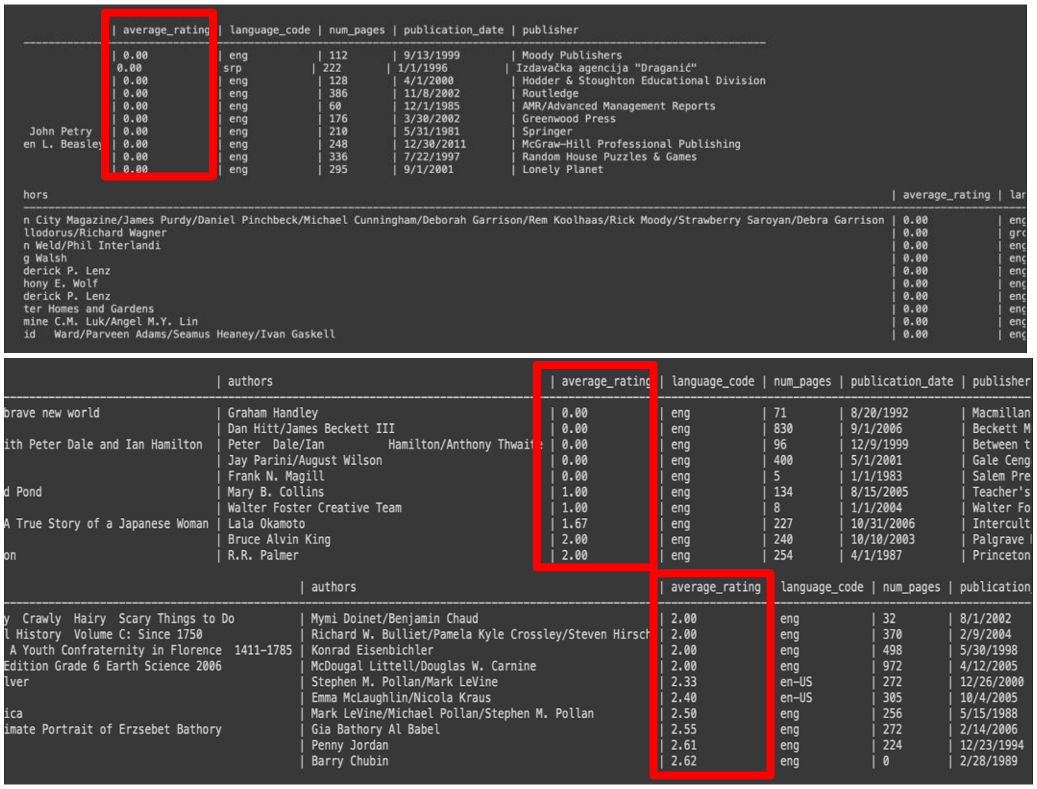
Actual output:



```
        ================================================
                              MENU
        ================================================
         1.  Load the books collection form CSV
         2.  Save the book collection to a CSV file
         3.  List all the books in the books collection
         4.  Show Info of a Book
         5.  Insert a book to the books collection
         6.  Delete the books from the books collection
         7.  Sort the books in the books collection
         8.  Search the books from the books collection
         9.  Staticstics of the book collection
         10. Update the data of specific book
         0. Exit the program
         CLR : Clear the Book Directory

Enter your choice: 7
Enter sorting group:
         1. BookID
         2. Title
         3. Average Ratings
         4. Number of pages
         5. Publication Date
         Enter your choice: 3
Enter the order:
         1. Ascending Order
         2. Descending Order
         Enter your choice: 1
The books have been successfully sorted.
```

Fig 36: Sort by Average Ratings - Asc order

Since this function doesn’t prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
             | average_rating | language_code | num_pages | publication_date | publisher
-----------------------------------------------------------------------------------------------------
             | 0.00           | eng           | 112       | 9/13/1999        | Moody Publishers
            0.00             srp            | 222       | 1/1/1996         | Izdavačka agencija "Draganić"
             | 0.00           | eng           | 128       | 4/1/2000         | Hodder & Stoughton Educational Division
             | 0.00           | eng           | 386       | 11/8/2002        | Routledge
             | 0.00           | eng           | 60        | 12/1/1985        | AMR/Advanced Management Reports
             | 0.00           | eng           | 176       | 3/30/2002        | Greenwood Press
 John Petry  | 0.00           | eng           | 210       | 5/31/1981        | Springer
en L. Beasley| 0.00           | eng           | 248       | 12/30/2011       | McGraw-Hill Professional Publishing
             | 0.00           | eng           | 336       | 7/22/1997        | Random House Puzzles & Games
             | 0.00           | eng           | 295       | 9/1/2001         | Lonely Planet

hors                                                                                                                    | average_rating | lar
-----------------------------------------------------------------------------------------------------------------------------------------------
n City Magazine/James Purdy/Daniel Pinchbeck/Michael Cunningham/Deborah Garrison/Rem Koolhaas/Rick Moody/Strawberry Saroyan/Debra Garrison | 0.00 | enc
llodorus/Richard Wagner                                                                                                 | 0.00           | grc
n Weld/Phil Interlandi                                                                                                  | 0.00           | enc
g Walsh                                                                                                                 | 0.00           | enc
derick P. Lenz                                                                                                          | 0.00           | enc
hony E. Wolf                                                                                                            | 0.00           | enc
derick P. Lenz                                                                                                          | 0.00           | enc
ter Homes and Gardens                                                                                                   | 0.00           | enc
mine C.M. Luk/Angel M.Y. Lin                                                                                            | 0.00           | enc
id   Ward/Parveen Adams/Seamus Heaney/Ivan Gaskell                                                                      | 0.00           | enc
```

```
                                 | authors                                 | average_rating | language_code | num_pages | publication_date | publisher
------------------------------------------------------------------------------------------------------------------------------------------------------
brave new world                  | Graham Handley                          | 0.00           | eng           | 71        | 8/20/1992        | Macmillan
                                 | Dan Hitt/James Beckett III              | 0.00           | eng           | 830       | 9/1/2006         | Beckett M
ith Peter Dale and Ian Hamilton  | Peter  Dale/Ian        Hamilton/Anthony Thwaite | 0.00   | eng           | 96        | 12/9/1999        | Between t
                                 | Jay Parini/August Wilson                | 0.00           | eng           | 400       | 5/1/2001         | Gale Ceng
                                 | Frank N. Magill                         | 0.00           | eng           | 5         | 1/1/1983         | Salem Pre
d Pond                           | Mary B. Collins                         | 1.00           | eng           | 134       | 8/15/2005        | Teacher's
                                 | Walter Foster Creative Team             | 1.00           | eng           | 8         | 1/1/2004         | Walter Fo
A True Story of a Japanese Woman | Lala Okamoto                            | 1.67           | eng           | 227       | 10/31/2006       | Intercult
                                 | Bruce Alvin King                        | 2.00           | eng           | 240       | 10/10/2003       | Palgrave
on                               | R.R. Palmer                             | 2.00           | eng           | 254       | 4/1/1987         | Princeton

                                       | authors                                          | average_rating   language_code | num_pages | publication
------------------------------------------------------------------------------------------------------------------------------------------------------
y  Crawly  Hairy  Scary Things to Do   | Mymi Doinet/Benjamin Chaud                       | 2.00             eng           | 32        | 8/1/2002
l History  Volume C: Since 1750        | Richard W. Bulliet/Pamela Kyle Crossley/Steven Hirsch | 2.00        eng           | 370       | 2/9/2004
 A Youth Confraternity in Florence  1411-1785 | Konrad Eisenbichler                       | 2.00             eng           | 498       | 5/30/1998
Edition Grade 6 Earth Science 2006     | McDougal Littell/Douglas W. Carnine              | 2.00             eng           | 972       | 4/12/2005
lver                                   | Stephen M. Pollan/Mark LeVine                    | 2.33             en-US         | 272       | 12/26/2000
                                       | Emma McLaughlin/Nicola Kraus                     | 2.40             en-US         | 305       | 10/4/2005
ica                                    | Mark LeVine/Michael Pollan/Stephen M. Pollan     | 2.50             eng           | 256       | 5/15/1988
imate Portrait of Erzsebet Bathory     | Gia Bathory Al Babel                             | 2.55             eng           | 272       | 2/14/2006
                                       | Penny Jordan                                     | 2.61             eng           | 224       | 12/23/1994
                                       | Barry Chubin                                     | 2.62             eng           | 0         | 2/28/1989
```

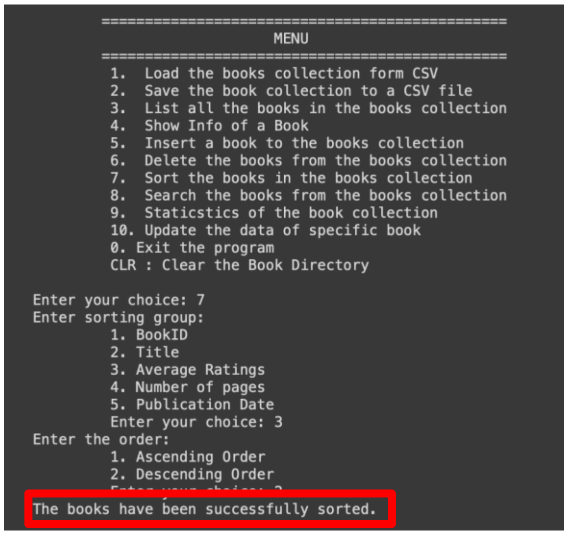
Fig 37: Sort by Average Ratings - Asc order - After calling list_books( )

Test Case 6: Sort by Average Ratings - Descending order

Input: User input number 3 for sorting group and number 2 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. “The books have been successfully sorted.” message should be prompted.

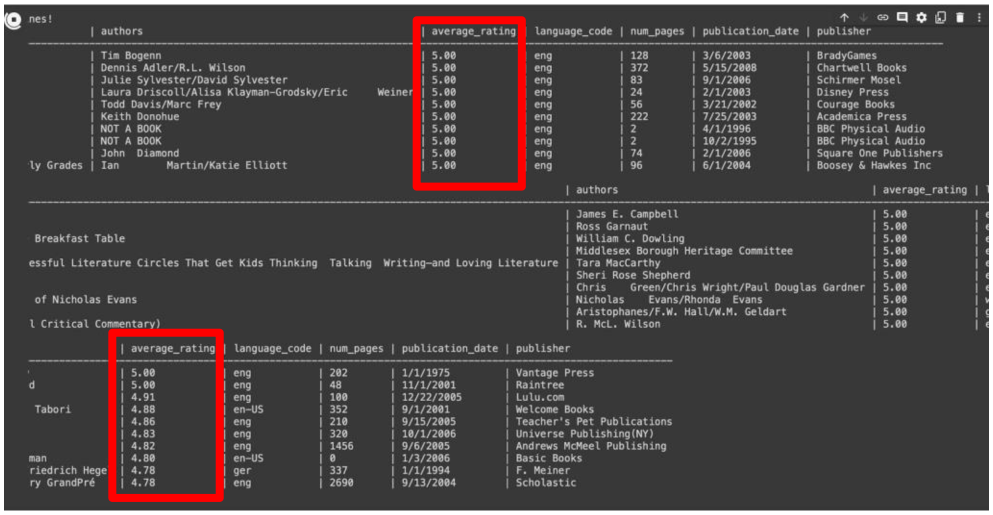
Actual output:



```
            ================================================
                                 MENU
            ================================================
             1.  Load the books collection form CSV
             2.  Save the book collection to a CSV file
             3.  List all the books in the books collection
             4.  Show Info of a Book
             5.  Insert a book to the books collection
             6.  Delete the books from the books collection
             7.  Sort the books in the books collection
             8.  Search the books from the books collection
             9.  Staticstics of the book collection
             10. Update the data of specific book
             0. Exit the program
             CLR : Clear the Book Directory

    Enter your choice: 7
    Enter sorting group:
             1. BookID
             2. Title
             3. Average Ratings
             4. Number of pages
             5. Publication Date
             Enter your choice: 3
    Enter the order:
             1. Ascending Order
             2. Descending Order
             Enter your choice: 2
    The books have been successfully sorted.
```

Fig38: Sort by Average Ratings - Desc order

Since this function doesn’t prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
nes!
         | authors                                     | average_rating | language_code | num_pages | publication_date | publisher
-----------------------------------------------------------------------------------------------------------------------------------
         | Tim Bogenn                                  | 5.00           | eng           | 128       | 3/6/2003         | BradyGames
         | Dennis Adler/R.L. Wilson                    | 5.00           | eng           | 372       | 5/15/2008        | Chartwell Books
         | Julie Sylvester/David Sylvester             | 5.00           | eng           | 83        | 9/1/2006         | Schirmer Mosel
         | Laura Driscoll/Alisa Klayman-Grodsky/Eric     Weiner | 5.00  | eng           | 24        | 2/1/2003         | Disney Press
         | Todd Davis/Marc Frey                        | 5.00           | eng           | 56        | 3/21/2002        | Courage Books
         | Keith Donohue                               | 5.00           | eng           | 222       | 7/25/2003        | Academica Press
         | NOT A BOOK                                  | 5.00           | eng           | 2         | 4/1/1996         | BBC Physical Audio
         | NOT A BOOK                                  | 5.00           | eng           | 2         | 10/2/1995        | BBC Physical Audio
         | John  Diamond                               | 5.00           | eng           | 74        | 2/1/2006         | Square One Publishers
ly Grades | Ian        Martin/Katie Elliott            | 5.00           | eng           | 96        | 6/1/2004         | Boosey & Hawkes Inc
                                                                                   | authors                                     | average_rating | 
------------------------------------------------------------------------------------------------------------------------------------------------------------
                                                                                   | James E. Campbell                           | 5.00           | e
                                                                                   | Ross Garnaut                                | 5.00           | e
 Breakfast Table                                                                   | William C. Dowling                          | 5.00           | e
                                                                                   | Middlesex Borough Heritage Committee        | 5.00           | e
essful Literature Circles That Get Kids Thinking  Talking  Writing—and Loving Literature | Tara MacCarthy                         | 5.00           | e
                                                                                   | Sheri Rose Shepherd                         | 5.00           | e
                                                                                   | Chris    Green/Chris Wright/Paul Douglas Gardner | 5.00      | e
 of Nicholas Evans                                                                 | Nicholas    Evans/Rhonda  Evans             | 5.00           | w
                                                                                   | Aristophanes/F.W. Hall/W.M. Geldart         | 5.00           | g
l Critical Commentary)                                                             | R. McL. Wilson                              | 5.00           | e
                | average_rating | language_code | num_pages | publication_date | publisher
----------------------------------------------------------------------------------------------
'               | 5.00           | eng           | 202       | 1/1/1975         | Vantage Press
d               | 5.00           | eng           | 48        | 11/1/2001        | Raintree
                | 4.91           | eng           | 100       | 12/22/2005       | Lulu.com
 Tabori         | 4.88           | en-US         | 352       | 9/1/2001         | Welcome Books
                | 4.86           | eng           | 210       | 9/15/2005        | Teacher's Pet Publications
                | 4.83           | eng           | 320       | 10/1/2006        | Universe Publishing(NY)
                | 4.82           | eng           | 1456      | 9/6/2005         | Andrews McMeel Publishing
man             | 4.80           | en-US         | 0         | 1/3/2006         | Basic Books
riedrich Hege   | 4.78           | ger           | 337       | 1/1/1994         | F. Meiner
ry GrandPré     | 4.78           | eng           | 2690      | 9/13/2004        | Scholastic
```

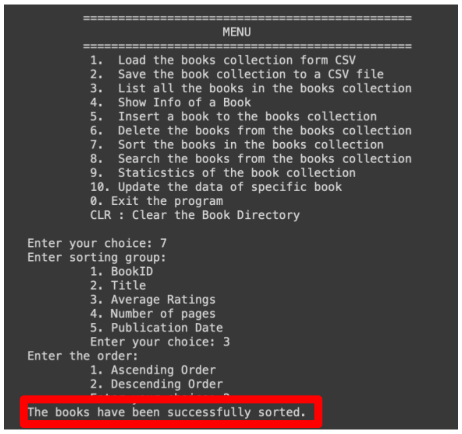
Fig 39: Sort by Average Ratings - Desc order - After calling list_books( )

Test Case 7: Sort by Number of pages - Ascending order

Input: User input number 4 for sorting group and number 1 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. “The books have been successfully sorted.” message should be prompted.

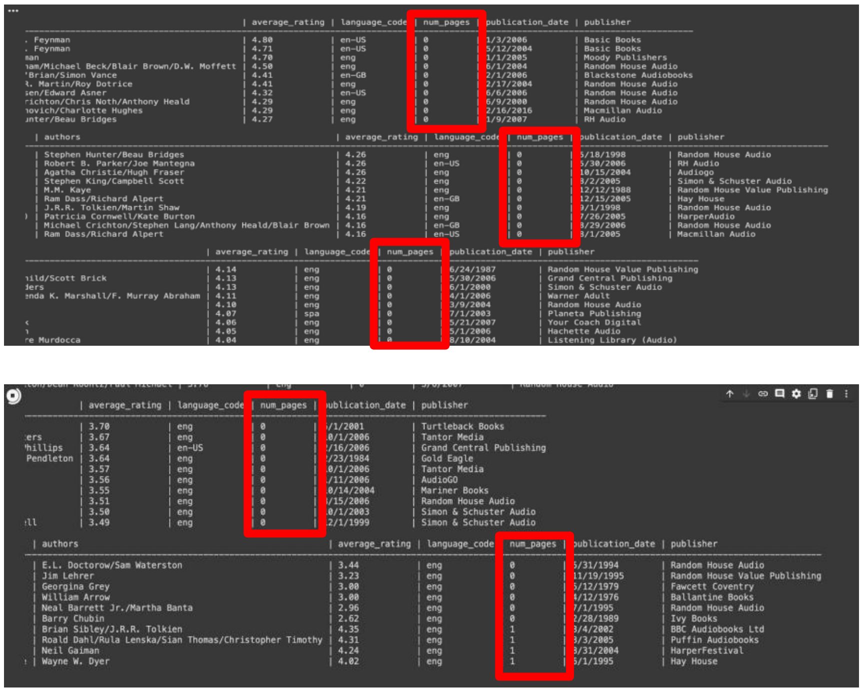
Actual output:



```
          ===============================================
                              MENU
          ===============================================
           1.  Load the books collection form CSV
           2.  Save the book collection to a CSV file
           3.  List all the books in the books collection
           4.  Show Info of a Book
           5.  Insert a book to the books collection
           6.  Delete the books from the books collection
           7.  Sort the books in the books collection
           8.  Search the books from the books collection
           9.  Staticstics of the book collection
           10. Update the data of specific book
           0. Exit the program
           CLR : Clear the Book Directory

Enter your choice: 7
Enter sorting group:
           1. BookID
           2. Title
           3. Average Ratings
           4. Number of pages
           5. Publication Date
           Enter your choice: 3
Enter the order:
           1. Ascending Order
           2. Descending Order
The books have been successfully sorted.
```

Fig 40: Sort by Number of pages - Asc order

Since this function doesn’t prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
                                          | average_rating | language_code | num_pages | publication_date | publisher
--------------------------------------------------------------------------------------------------------------------------
. Feynman                                 | 4.80           | en-US         | 0         | 1/3/2006         | Basic Books
. Feynman                                 | 4.71           | en-US         | 0         | 5/12/2004        | Basic Books
man                                       | 4.70           | eng           | 0         | 1/1/2005         | Moody Publishers
ham/Michael Beck/Blair Brown/D.W. Moffett | 4.50           | eng           | 0         | 6/1/2004         | Random House Audio
'Brian/Simon Vance                        | 4.41           | en-GB         | 0         | 2/1/2006         | Blackstone Audiobooks
R. Martin/Roy Dotrice                     | 4.41           | eng           | 0         | 2/17/2004        | Random House Audio
sen/Edward Asner                          | 4.32           | en-US         | 0         | 6/6/2006         | Random House Audio
richton/Chris Noth/Anthony Heald          | 4.29           | eng           | 0         | 6/9/2000         | Random House Audio
novich/Charlotte Hughes                   | 4.29           | eng           | 0         | 2/16/2016        | Macmillan Audio
unter/Beau Bridges                        | 4.27           | eng           | 0         | 1/9/2007         | RH Audio
  | authors                                                  | average_rating | language_code | num_pages | publication_date | publisher
----------------------------------------------------------------------------------------------------------------------------------------------
  | Stephen Hunter/Beau Bridges                              | 4.26           | eng           | 0         | 5/18/1998        | Random House Audio
  | Robert B. Parker/Joe Mantegna                            | 4.26           | en-US         | 0         | 5/30/2006        | RH Audio
  | Agatha Christie/Hugh Fraser                              | 4.26           | eng           | 0         | 10/15/2004       | Audiogo
  | Stephen King/Campbell Scott                              | 4.22           | eng           | 0         | 3/2/2005         | Simon & Schuster Audio
  | M.M. Kaye                                                | 4.21           | eng           | 0         | 12/12/1988       | Random House Value Publishing
  | Ram Dass/Richard Alpert                                  | 4.21           | en-GB         | 0         | 12/15/2005       | Hay House
  | J.R.R. Tolkien/Martin Shaw                               | 4.19           | eng           | 0         | 9/1/1998         | Random House Audio
) | Patricia Cornwell/Kate Burton                            | 4.16           | eng           | 0         | 7/26/2005        | HarperAudio
  | Michael Crichton/Stephen Lang/Anthony Heald/Blair Brown | 4.16           | en-GB         | 0         | 3/29/2006        | Random House Audio
  | Ram Dass/Richard Alpert                                  | 4.16           | en-US         | 0         | 3/1/2005         | Macmillan Audio
                                       | average_rating | language_code | num_pages | publication_date | publisher
----------------------------------------------------------------------------------------------------------------
                                       | 4.14           | eng           | 0         | 6/24/1987        | Random House Value Publishing
ild/Scott Brick                        | 4.13           | eng           | 0         | 5/30/2006        | Grand Central Publishing
ders                                   | 4.13           | eng           | 0         | 6/1/2000         | Simon & Schuster Audio
enda K. Marshall/F. Murray Abraham     | 4.11           | eng           | 0         | 4/1/2006         | Warner Adult
                                       | 4.10           | eng           | 0         | 3/9/2004         | Random House Audio
                                       | 4.07           | spa           | 0         | 7/1/2003         | Planeta Publishing
<                                      | 4.06           | eng           | 0         | 5/21/2007        | Your Coach Digital
1                                      | 4.05           | eng           | 0         | 5/1/2006         | Hachette Audio
re Murdocca                            | 4.04           | eng           | 0         | 8/10/2004        | Listening Library (Audio)
```

```
               | average_rating | language_code | num_pages | publication_date | publisher
-----------------------------------------------------------------------------------------
               | 3.70           | eng           | 0         | 5/1/2001         | Turtleback Books
ers            | 3.67           | eng           | 0         | 10/1/2006        | Tantor Media
hillips        | 3.64           | en-US         | 0         | 2/16/2006        | Grand Central Publishing
 Pendleton     | 3.64           | eng           | 0         | 2/23/1984        | Gold Eagle
               | 3.57           | eng           | 0         | 10/1/2006        | Tantor Media
               | 3.56           | eng           | 0         | 1/11/2006        | AudioGO
               | 3.55           | eng           | 0         | 10/14/2004       | Mariner Books
               | 3.51           | eng           | 0         | 3/15/2006        | Random House Audio
               | 3.50           | eng           | 0         | 10/1/2003        | Simon & Schuster Audio
ll             | 3.49           | eng           | 0         | 12/1/1999        | Simon & Schuster Audio
  | authors                                                   | average_rating | language_code | num_pages | publication_date | publisher
-----------------------------------------------------------------------------------------------------------------------------------------------
  | E.L. Doctorow/Sam Waterston                               | 3.44           | eng           | 0         | 5/31/1994        | Random House Audio
  | Jim Lehrer                                                | 3.23           | eng           | 0         | 11/19/1995       | Random House Value Publishing
  | Georgina Grey                                             | 3.00           | eng           | 0         | 5/12/1979        | Fawcett Coventry
  | William Arrow                                             | 3.00           | eng           | 0         | 4/12/1976        | Ballantine Books
  | Neal Barrett Jr./Martha Banta                             | 2.96           | eng           | 0         | 7/1/1995         | Random House Audio
  | Barry Chubin                                              | 2.62           | eng           | 0         | 2/28/1989        | Ivy Books
  | Brian Sibley/J.R.R. Tolkien                               | 4.35           | eng           | 1         | 3/4/2002         | BBC Audiobooks Ltd
  | Roald Dahl/Rula Lenska/Sian Thomas/Christopher Timothy    | 4.31           | eng           | 1         | 3/3/2005         | Puffin Audiobooks
  | Neil Gaiman                                               | 4.24           | eng           | 1         | 3/31/2004        | HarperFestival
: | Wayne W. Dyer                                             | 4.02           | eng           | 1         | 5/1/1995         | Hay House
```

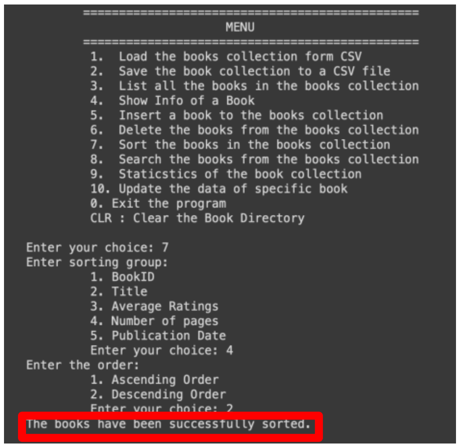
Fig 41: Sort by Number of pages - Asc order - After calling list_books( )

Test Case 8: Sort by Number of pages - Descending order

Input: User input number 4 for sorting group and number 2 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. “The books have been successfully sorted.” message should be prompted.

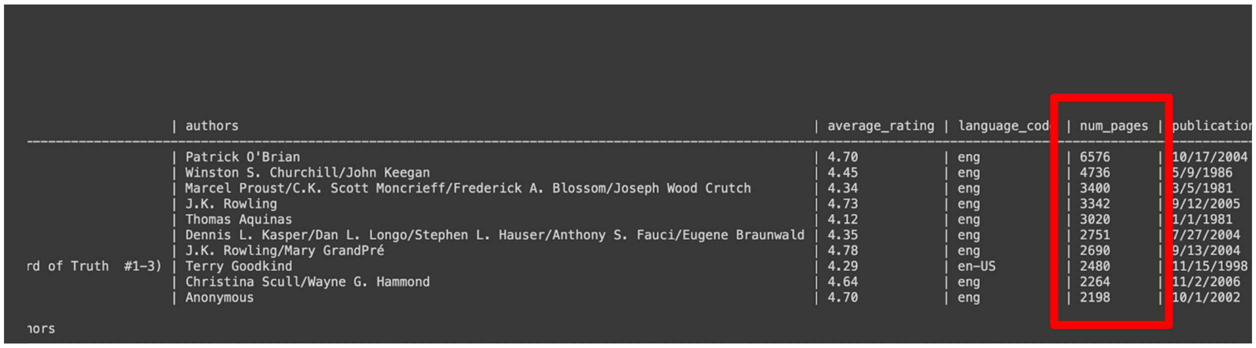
Actual output:



```
         ==============================================
                              MENU
         ==============================================
          1.  Load the books collection form CSV
          2.  Save the book collection to a CSV file
          3.  List all the books in the books collection
          4.  Show Info of a Book
          5.  Insert a book to the books collection
          6.  Delete the books from the books collection
          7.  Sort the books in the books collection
          8.  Search the books from the books collection
          9.  Staticstics of the book collection
          10. Update the data of specific book
          0. Exit the program
          CLR : Clear the Book Directory

Enter your choice: 7
Enter sorting group:
          1. BookID
          2. Title
          3. Average Ratings
          4. Number of pages
          5. Publication Date
          Enter your choice: 4
Enter the order:
          1. Ascending Order
          2. Descending Order
          Enter your choice: 2
The books have been successfully sorted.
```

Fig 42: Sort by Number of pages- Desc order

Since this function doesn’t prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
                    | authors                                                                | average_rating | language_cod | num_pages | publicatior
--------------------------------------------------------------------------------------------------------------------------------------------------
                    | Patrick O'Brian                                                        | 4.70           | eng          | 6576      | 10/17/2004
                    | Winston S. Churchill/John Keegan                                       | 4.45           | eng          | 4736      | 5/9/1986
                    | Marcel Proust/C.K. Scott Moncrieff/Frederick A. Blossom/Joseph Wood Crutch | 4.34       | eng          | 3400      | 3/5/1981
                    | J.K. Rowling                                                           | 4.73           | eng          | 3342      | 9/12/2005
                    | Thomas Aquinas                                                         | 4.12           | eng          | 3020      | 1/1/1981
                    | Dennis L. Kasper/Dan L. Longo/Stephen L. Hauser/Anthony S. Fauci/Eugene Braunwald | 4.35 | eng          | 2751      | 7/27/2004
                    | J.K. Rowling/Mary GrandPré                                             | 4.78           | eng          | 2690      | 9/13/2004
rd of Truth  #1-3)  | Terry Goodkind                                                         | 4.29           | en-US        | 2480      | 11/15/1998
                    | Christina Scull/Wayne G. Hammond                                       | 4.64           | eng          | 2264      | 11/2/2006
                    | Anonymous                                                              | 4.70           | eng          | 2198      | 10/1/2002

ɿors
```

Fig 43: Sort by Number of pages - Desc order - After calling list_books( )

Test Case 9: Sort by Publication Date - Ascending order

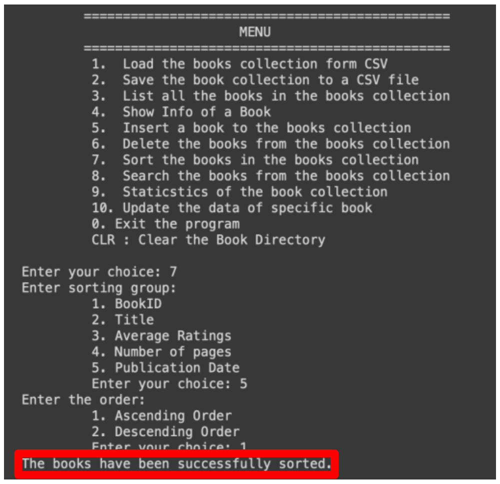
Helper Function : convert_to_datetime(book)

filter_valid_dates(books):

Input: User input number 5 for sorting group and number 1 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. "The books have been successfully sorted." message should be prompted.

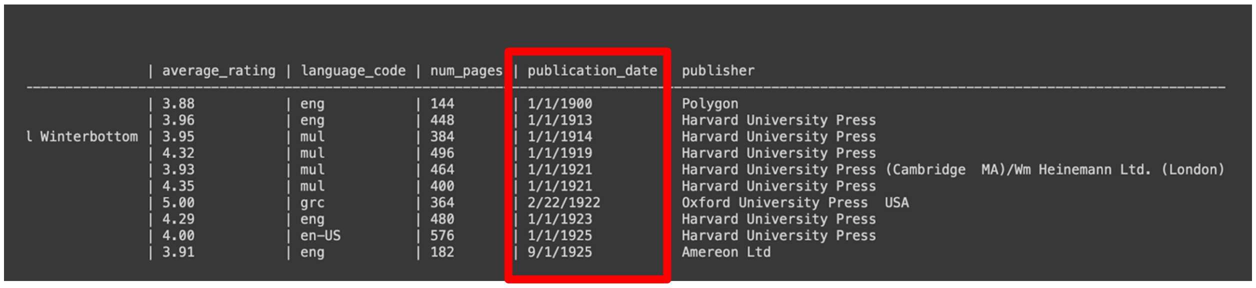
Actual output:



```
        ================================================
                              MENU
        ================================================
         1.  Load the books collection form CSV
         2.  Save the book collection to a CSV file
         3.  List all the books in the books collection
         4.  Show Info of a Book
         5.  Insert a book to the books collection
         6.  Delete the books from the books collection
         7.  Sort the books in the books collection
         8.  Search the books from the books collection
         9.  Staticstics of the book collection
         10. Update the data of specific book
         0. Exit the program
         CLR : Clear the Book Directory

Enter your choice: 7
Enter sorting group:
         1. BookID
         2. Title
         3. Average Ratings
         4. Number of pages
         5. Publication Date
         Enter your choice: 5
Enter the order:
         1. Ascending Order
         2. Descending Order
         Enter your choice: 1
The books have been successfully sorted.
```

Fig 44: Sort by Publication Date - Asc order

Since this function doesn't prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
                 | average_rating | language_code | num_pages | publication_date  publisher
------------------------------------------------------------------------------------------------------------------------------------------
                 | 3.88           | eng           | 144       | 1/1/1900          Polygon
                 | 3.96           | eng           | 448       | 1/1/1913          Harvard University Press
l Winterbottom   | 3.95           | mul           | 384       | 1/1/1914          Harvard University Press
                 | 4.32           | mul           | 496       | 1/1/1919          Harvard University Press
                 | 3.93           | mul           | 464       | 1/1/1921          Harvard University Press (Cambridge  MA)/Wm Heinemann Ltd. (London)
                 | 4.35           | mul           | 400       | 1/1/1921          Harvard University Press
                 | 5.00           | grc           | 364       | 2/22/1922         Oxford University Press  USA
                 | 4.29           | eng           | 480       | 1/1/1923          Harvard University Press
                 | 4.00           | en-US         | 576       | 1/1/1925          Harvard University Press
                 | 3.91           | eng           | 182       | 9/1/1925          Amereon Ltd
```

Fig 45: Sort by Publication Date - Asc order - After calling list_books( )

Test Case 10: Sort by Publication Date - Descending order

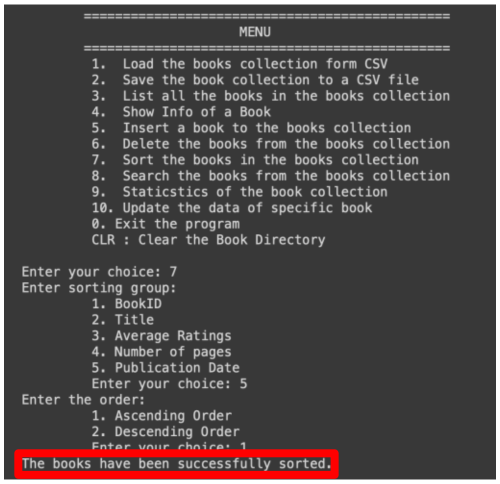
Helper Function : convert_to_datetime(book)

filter_valid_dates(books):

Input: User input number 5 for sorting group and number 2 for order.

Expected output: active_books global variable should be updated in ascending order of the bookID. “The books have been successfully sorted.” message should be prompted.

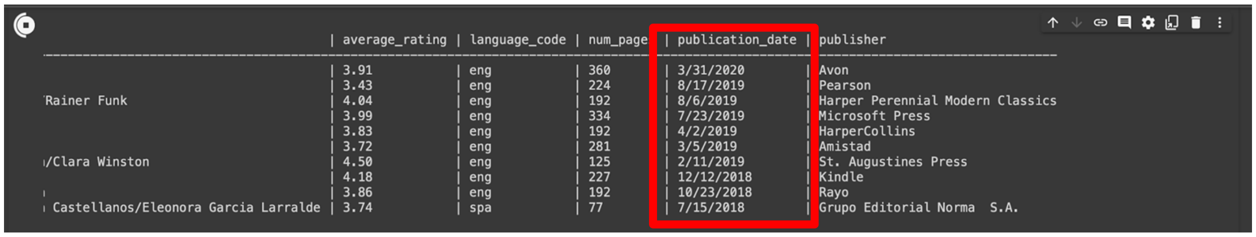
Actual output:



```
        ===============================================
                             MENU
        ===============================================
         1.  Load the books collection form CSV
         2.  Save the book collection to a CSV file
         3.  List all the books in the books collection
         4.  Show Info of a Book
         5.  Insert a book to the books collection
         6.  Delete the books from the books collection
         7.  Sort the books in the books collection
         8.  Search the books from the books collection
         9.  Staticstics of the book collection
         10. Update the data of specific book
         0. Exit the program
         CLR : Clear the Book Directory

Enter your choice: 7
Enter sorting group:
         1. BookID
         2. Title
         3. Average Ratings
         4. Number of pages
         5. Publication Date
         Enter your choice: 5
Enter the order:
         1. Ascending Order
         2. Descending Order
         Enter your choice: 1
The books have been successfully sorted.
```

Fig 46: Sort by Publication Date - Desc order

Since this function doesn’t prompt sorted books to the console, using the list_books( ) function we can demonstrate the result.



```
                                   | average_rating | language_code | num_page  | publication_date | publisher
-------------------------------------------------------------------------------------------------------------------------------------
                                   | 3.91           | eng           | 360       | 3/31/2020        | Avon
                                   | 3.43           | eng           | 224       | 8/17/2019        | Pearson
Rainer Funk                        | 4.04           | eng           | 192       | 8/6/2019         | Harper Perennial Modern Classics
                                   | 3.99           | eng           | 334       | 7/23/2019        | Microsoft Press
                                   | 3.83           | eng           | 192       | 4/2/2019         | HarperCollins
                                   | 3.72           | eng           | 281       | 3/5/2019         | Amistad
/Clara Winston                     | 4.50           | eng           | 125       | 2/11/2019        | St. Augustines Press
                                   | 4.18           | eng           | 227       | 12/12/2018       | Kindle
                                   | 3.86           | eng           | 192       | 10/23/2018       | Rayo
 Castellanos/Eleonora Garcia Larralde | 3.74        | spa           | 77        | 7/15/2018        | Grupo Editorial Norma  S.A.
```

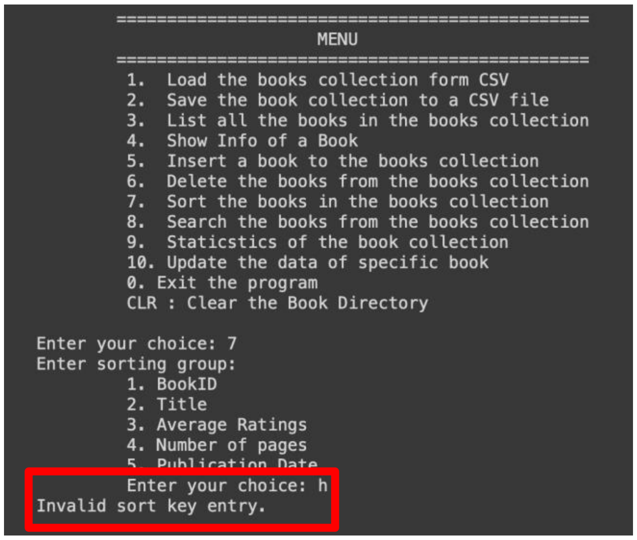
Fig 47: Sort by Publication Date - Desc order - After calling list_books( )

Test Case 11: Sort - Invalid sort key

Input: User input letter ‘h’ for sorting group.

Expected output: “Invalid sort key entry.” message should be prompted. Text-based menu should be prompted.

Actual output:



```
           ================================================
                                MENU
           ================================================
            1.  Load the books collection form CSV
            2.  Save the book collection to a CSV file
            3.  List all the books in the books collection
            4.  Show Info of a Book
            5.  Insert a book to the books collection
            6.  Delete the books from the books collection
            7.  Sort the books in the books collection
            8.  Search the books from the books collection
            9.  Staticstics of the book collection
            10. Update the data of specific book
            0. Exit the program
            CLR : Clear the Book Directory

   Enter your choice: 7
   Enter sorting group:
            1. BookID
            2. Title
            3. Average Ratings
            4. Number of pages
            5. Publication Date
            Enter your choice: h
   Invalid sort key entry.
```

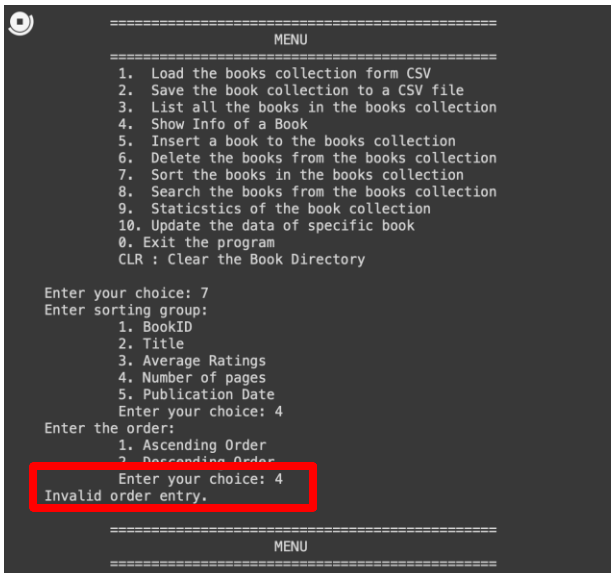
Fig 48: Sort - Invalid sort key

Test Case 11: Sort - Invalid order

Input: User input letter 4 for sorting group and input number 4 for the order..

Expected output: “Invalid order entry.” message should be prompted. Text-based menu should be prompted.

Actual output:



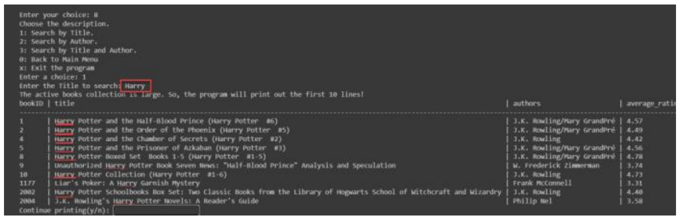
```
           ================================================
                                MENU
           ================================================
            1.  Load the books collection form CSV
            2.  Save the book collection to a CSV file
            3.  List all the books in the books collection
            4.  Show Info of a Book
            5.  Insert a book to the books collection
            6.  Delete the books from the books collection
            7.  Sort the books in the books collection
            8.  Search the books from the books collection
            9.  Staticstics of the book collection
            10. Update the data of specific book
            0. Exit the program
            CLR : Clear the Book Directory

   Enter your choice: 7
   Enter sorting group:
            1. BookID
            2. Title
            3. Average Ratings
            4. Number of pages
            5. Publication Date
            Enter your choice: 4
   Enter the order:
            1. Ascending Order
            2. Descending Order
            Enter your choice: 4
   Invalid order entry.

           ================================================
                                MENU
           ================================================
```

Fig 49: Sort - Invalid order

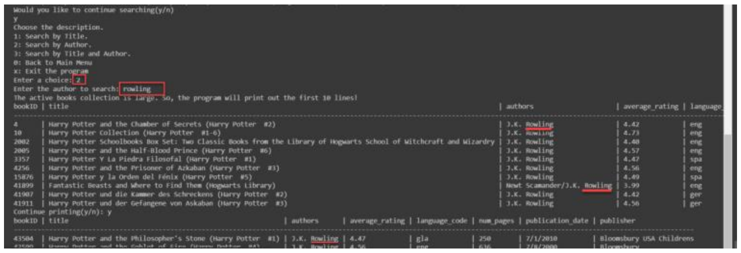
Function 9: search_books( )

Test Case 1: Valid Input

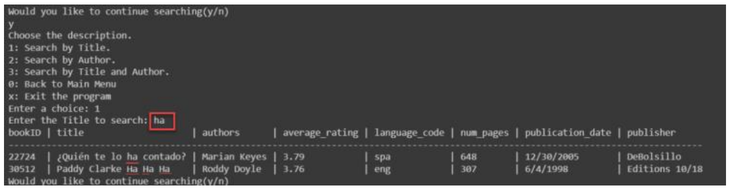
Actual Output:



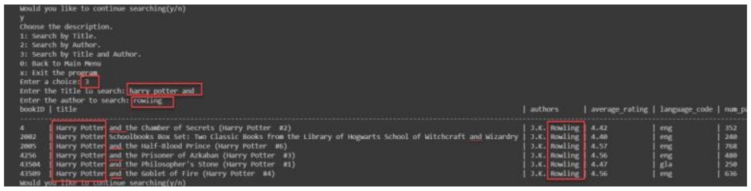
```
Enter your choice: 8
Choose the description.
1: Search by Title.
2: Search by Author.
3: Search by Title and Author.
0: Back to Main Menu
x: Exit the program
Enter a choice: 1
Enter the Title to search: Harry
The active books collection is large. So, the program will print out the first 10 lines!
bookID | title                                                                                    | authors                 | average_ratin
-------------------------------------------------------------------------------------------------------------------------------------------
1      | Harry Potter and the Half-Blood Prince (Harry Potter  #6)                                | J.K. Rowling/Mary GrandPré | 4.57
2      | Harry Potter and the Order of the Phoenix (Harry Potter  #5)                             | J.K. Rowling/Mary GrandPré | 4.49
4      | Harry Potter and the Chamber of Secrets (Harry Potter  #2)                               | J.K. Rowling            | 4.42
5      | Harry Potter and the Prisoner of Azkaban (Harry Potter  #3)                              | J.K. Rowling/Mary GrandPré | 4.56
8      | Harry Potter Boxed Set  Books 1-5 (Harry Potter  #1-5)                                   | J.K. Rowling/Mary GrandPré | 4.78
9      | Unauthorized Harry Potter Book Seven News: "Half-Blood Prince" Analysis and Speculation | W. Frederick Zimmerman  | 3.74
10     | Harry Potter Collection (Harry Potter  #1-6)                                             | J.K. Rowling            | 4.73
1177   | Liar's Poker: A Harry Garnish Mystery                                                    | Frank McConnell         | 3.31
2002   | Harry Potter Schoolbooks Box Set: Two Classic Books from the Library of Hogwarts School of Witchcraft and Wizardry | J.K. Rowling | 4.40
2004   | J.K. Rowling's Harry Potter Novels: A Reader's Guide                                     | Philip Nel              | 3.58
Continue printing(y/n):
```



```
Would you like to continue searching(y/n)
y
Choose the description.
1: Search by Title.
2: Search by Author.
3: Search by Title and Author.
0: Back to Main Menu
x: Exit the program
Enter a choice: 2
Enter the author to search: rowling
The active books collection is large. So, the program will print out the first 10 lines!
bookID | title                                                                                    | authors                   | average_rating | language_
--------------------------------------------------------------------------------------------------------------------------------------------------------------
4      | Harry Potter and the Chamber of Secrets (Harry Potter  #2)                               | J.K. Rowling              | 4.42           | eng
10     | Harry Potter Collection (Harry Potter  #1-6)                                             | J.K. Rowling              | 4.73           | eng
2002   | Harry Potter Schoolbooks Box Set: Two Classic Books from the Library of Hogwarts School of Witchcraft and Wizardry | J.K. Rowling | 4.40 | eng
2005   | Harry Potter and the Half-Blood Prince (Harry Potter  #6)                                | J.K. Rowling              | 4.57           | eng
3357   | Harry Potter Y La Piedra Filosofal (Harry Potter  #1)                                    | J.K. Rowling              | 4.47           | spa
4256   | Harry Potter and the Prisoner of Azkaban (Harry Potter  #3)                              | J.K. Rowling              | 4.56           | eng
15876  | Harry Potter y la Orden del Fénix (Harry Potter  #5)                                     | J.K. Rowling              | 4.49           | spa
41899  | Fantastic Beasts and Where to Find Them (Hogwarts Library)                               | Newt Scamander/J.K. Rowling | 3.99         | eng
41907  | Harry Potter und die Kammer des Schreckens (Harry Potter  #2)                            | J.K. Rowling              | 4.42           | ger
41911  | Harry Potter und der Gefangene von Askaban (Harry Potter  #3)                            | J.K. Rowling              | 4.56           | ger
Continue printing(y/n): y
bookID | title                                                   | authors      | average_rating | language_code | num_pages | publication_date | publisher
------------------------------------------------------------------------------------------------------------------------------------------------------
43504  | Harry Potter and the Philosopher's Stone (Harry Potter  #1) | J.K. Rowling | 4.47        | gla           | 250       | 7/1/2010         | Bloomsbury USA Childrens
```



```
Would you like to continue searching(y/n)
y
Choose the description.
1: Search by Title.
2: Search by Author.
3: Search by Title and Author.
0: Back to Main Menu
x: Exit the program
Enter a choice: 1
Enter the Title to search: ha
bookID | title                    | authors       | average_rating | language_code | num_pages | publication_date | publisher
---------------------------------------------------------------------------------------------------------------------------
22724  | ¿Quién te lo ha contado? | Marian Keyes  | 3.79           | spa           | 648       | 12/30/2005       | DeBolsillo
30512  | Paddy Clarke Ha Ha Ha    | Roddy Doyle   | 3.76           | eng           | 307       | 6/4/1998         | Editions 10/18
Would you like to continue searching(y/n)
```



```
Would you like to continue searching(y/n)
y
Choose the description.
1: Search by Title.
2: Search by Author.
3: Search by Title and Author.
0: Back to Main Menu
x: Exit the program
Enter a choice: 3
Enter the Title to search: harry potter and
Enter the author to search: rowling
bookID | title                                                                                    | authors      | average_rating | language_code | num_p
-------------------------------------------------------------------------------------------------------------------------------------------------------
4      | Harry Potter and the Chamber of Secrets (Harry Potter  #2)                               | J.K. Rowling | 4.42           | eng           | 352
2002   | Harry Potter Schoolbooks Box Set: Two Classic Books from the Library of Hogwarts School of Witchcraft and Wizardry | J.K. Rowling | 4.40 | eng | 240
2005   | Harry Potter and the Half-Blood Prince (Harry Potter  #6)                                | J.K. Rowling | 4.57           | eng           | 768
4256   | Harry Potter and the Prisoner of Azkaban (Harry Potter  #3)                              | J.K. Rowling | 4.56           | eng           | 480
43504  | Harry Potter and the Philosopher's Stone (Harry Potter  #1)                              | J.K. Rowling | 4.47           | gla           | 250
43509  | Harry Potter and the Goblet of Fire (Harry Potter  #4)                                   | J.K. Rowling | 4.56           | eng           | 636
Would you like to continue searching(y/n)
```

Fig 50: search_books - Valid Input

Input: Users need to choose the options(title, author or both) to search the books.

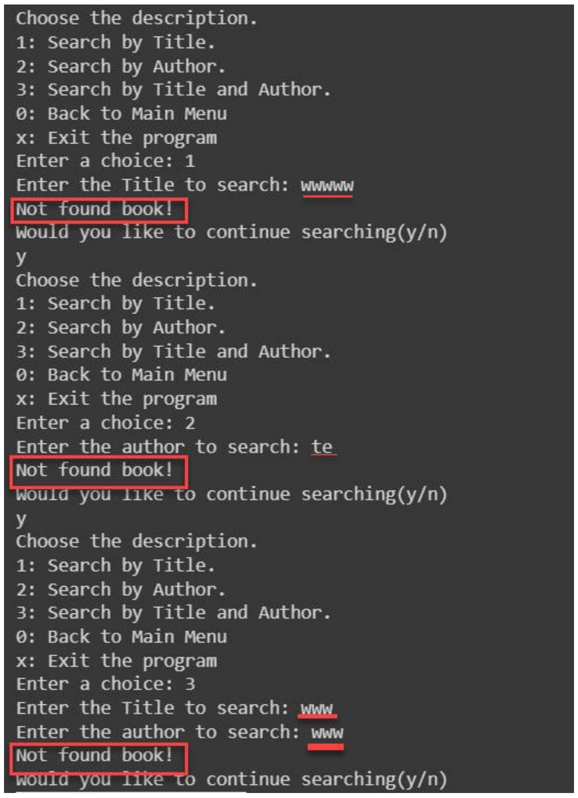
Expected Output: The list of the books will appear based on the user's choice. Even though the user input the lowercase, if the searching books exist, the output will be displayed on all of the listed books which are matched with the input data. The program will also search each word input data.

Test Case 2: Invalid Input

Input: Users need to choose the options to search the books.

Expected Output: The Output "Not Found Book!" will be displayed when the user types nonexistent data.

Actual Output:



```
Choose the description.
1: Search by Title.
2: Search by Author.
3: Search by Title and Author.
0: Back to Main Menu
x: Exit the program
Enter a choice: 1
Enter the Title to search: wwwww
Not found book!
Would you like to continue searching(y/n)
y
Choose the description.
1: Search by Title.
2: Search by Author.
3: Search by Title and Author.
0: Back to Main Menu
x: Exit the program
Enter a choice: 2
Enter the author to search: te
Not found book!
Would you like to continue searching(y/n)
y
Choose the description.
1: Search by Title.
2: Search by Author.
3: Search by Title and Author.
0: Back to Main Menu
x: Exit the program
Enter a choice: 3
Enter the Title to search: www
Enter the author to search: www
Not found book!
Would you like to continue searching(y/n)
```

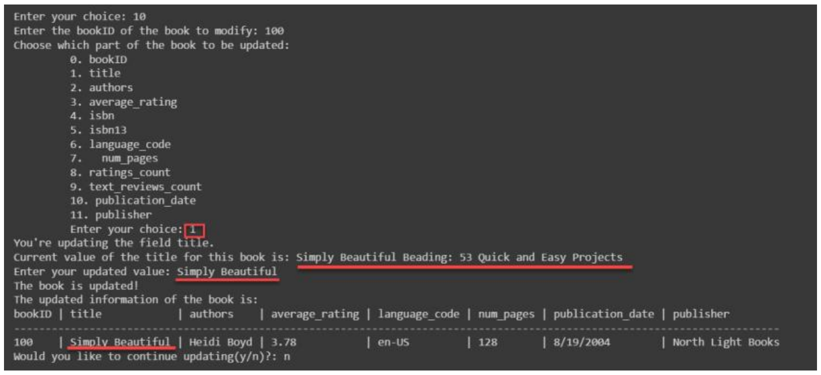
Fig 51: search_books - Invalid Input

Function 10: update_books( ):

Input: Users need to choose the categories to update the books.

Expected Output: After updating the information of the book, a successful message will appear. And also the update book data will be displayed with the well-formatted table.

Actual Output:

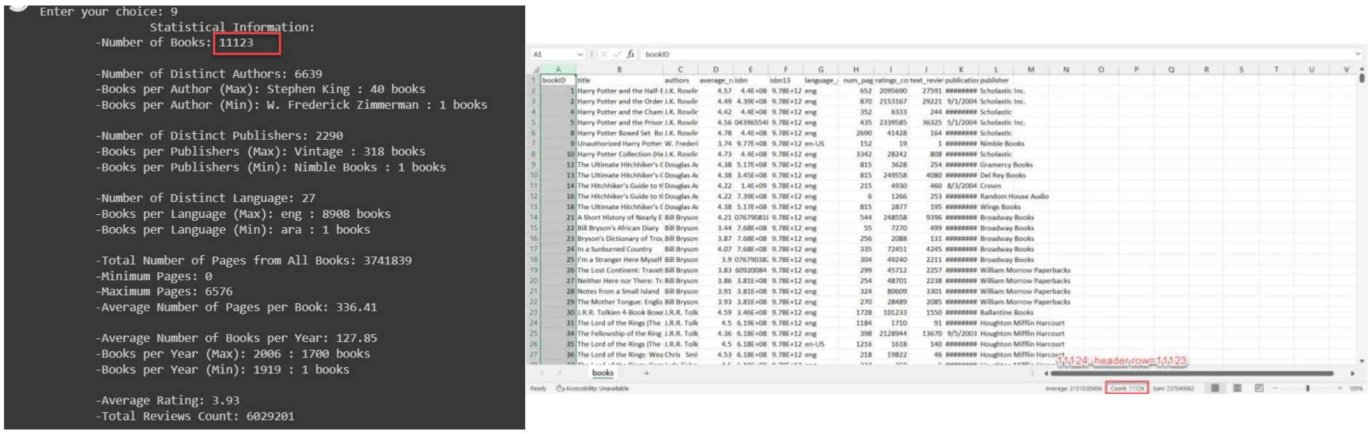


```
Enter your choice: 10
Enter the bookID of the book to modify: 100
Choose which part of the book to be updated:
         0. bookID
         1. title
         2. authors
         3. average_rating
         4. isbn
         5. isbn13
         6. language_code
         7.   num_pages
         8. ratings_count
         9. text_reviews_count
         10. publication_date
         11. publisher
         Enter your choice: 1
You're updating the field title.
Current value of the title for this book is: Simply Beautiful Beading: 53 Quick and Easy Projects
Enter your updated value: Simply Beautiful
The book is updated!
The updated information of the book is:
bookID | title            | authors     | average_rating | language_code | num_pages | publication_date | publisher
------------------------------------------------------------------------------------------------------------------
100    | Simply Beautiful | Heidi Boyd | 3.78            | en-US         | 128       | 8/19/2004        | North Light Books
Would you like to continue updating(y/n)?: n
```

Fig 52: update_books - Test Case

Function 11: calculate_statistics(books_list)

Actual Output:



```
Enter your choice: 9
                  Statistical Information:
         -Number of Books: 11123

         -Number of Distinct Authors: 6639
         -Books per Author (Max): Stephen King : 40 books
         -Books per Author (Min): W. Frederick Zimmerman : 1 books

         -Number of Distinct Publishers: 2290
         -Books per Publishers (Max): Vintage : 318 books
         -Books per Publishers (Min): Nimble Books : 1 books

         -Number of Distinct Language: 27
         -Books per Language (Max): eng : 8908 books
         -Books per Language (Min): ara : 1 books

         -Total Number of Pages from All Books: 3741839
         -Minimum Pages: 0
         -Maximum Pages: 6576
         -Average Number of Pages per Book: 336.41

         -Average Number of Books per Year: 127.85
         -Books per Year (Max): 2006 : 1700 books
         -Books per Year (Min): 1919 : 1 books

         -Average Rating: 3.93
         -Total Reviews Count: 6029201
```

bookID	title	authors	average_r	isbn	isbn13	language_	num_pag	ratings_co	text_revie	publicatior	publisher
1	Harry Potter and the Half-E	J.K. Rowlir	4.57	4.4E+08	9.78E+12	eng	652	2095690	27591	########	Scholastic Inc.
2	Harry Potter and the Order	J.K. Rowlir	4.49	4.39E+08	9.78E+12	eng	870	2153167	29221	9/1/2004	Scholastic Inc.
4	Harry Potter and the Cham	J.K. Rowlir	4.42	4.4E+08	9.78E+12	eng	352	6333	244	########	Scholastic
5	Harry Potter and the Prisor	J.K. Rowlir	4.56	04396554	9.78E+12	eng	435	2339585	36325	5/1/2004	Scholastic Inc.
8	Harry Potter Boxed Set Bo	J.K. Rowlir	4.78	4.4E+08	9.78E+12	eng	2690	41428	164	########	Scholastic
9	Unauthorized Harry Potter	W. Frederi	3.74	9.77E+08	9.78E+12	en-US	152	19	1	########	Nimble Books
10	Harry Potter Collection (Ha	J.K. Rowlir	4.73	4.4E+08	9.78E+12	eng	3342	28242	808	########	Scholastic
12	The Ultimate Hitchhiker's C	Douglas Ac	4.38	5.17E+08	9.78E+12	eng	815	3628	254	########	Gramercy Books
13	The Ultimate Hitchhiker's C	Douglas Ac	4.38	3.45E+08	9.78E+12	eng	815	249558	4080	########	Del Rey Books
14	The Hitchhiker's Guide to tl	Douglas Ac	4.22	1.4E+09	9.78E+12	eng	215	4930	460	8/3/2004	Crown
16	The Hitchhiker's Guide to tl	Douglas Ac	4.22	7.39E+08	9.78E+12	eng	6	1266	253	########	Random House Audio
18	The Ultimate Hitchhiker's C	Douglas Ac	4.38	5.17E+08	9.78E+12	eng	815	2877	195	########	Wings Books
21	A Short History of Nearly E	Bill Bryson	4.21	07679081	9.78E+12	eng	544	248558	9396	########	Broadway Books
22	Bill Bryson's African Diary	Bill Bryson	3.44	7.68E+08	9.78E+12	eng	55	7270	499	########	Broadway Books
23	Bryson's Dictionary of Trou	Bill Bryson	3.87	7.68E+08	9.78E+12	eng	256	2088	131	########	Broadway Books
24	In a Sunburned Country	Bill Bryson	4.07	7.68E+08	9.78E+12	eng	335	72451	4245	########	Broadway Books
25	I'm a Stranger Here Myself	Bill Bryson	3.9	07679038	9.78E+12	eng	304	49240	2211	########	Broadway Books
26	The Lost Continent: Travel	Bill Bryson	3.83	60920084	9.78E+12	eng	299	45712	2257	########	William Morrow Paperbacks
27	Neither Here nor There: Tr	Bill Bryson	3.86	3.81E+08	9.78E+12	eng	254	48701	2238	########	William Morrow Paperbacks
28	Notes from a Small Island	Bill Bryson	3.91	3.81E+08	9.78E+12	eng	324	80609	3301	########	William Morrow Paperbacks
29	The Mother Tongue: Englis	Bill Bryson	3.93	3.81E+08	9.78E+12	eng	270	28489	2085	########	William Morrow Paperbacks
30	J.R.R. Tolkien 4-Book Boxe	J.R.R. Tolk	4.59	3.46E+08	9.78E+12	eng	1728	101233	1550	########	Ballantine Books
31	The Lord of the Rings (The	J.R.R. Tolk	4.5	6.19E+08	9.78E+12	eng	1184	1710	91	########	Houghton Mifflin Harcourt
34	The Fellowship of the Ring	J.R.R. Tolk	4.36	6.18E+08	9.78E+12	eng	398	2128944	13670	9/5/2003	Houghton Mifflin Harcourt
35	The Lord of the Rings (The	J.R.R. Tolk	4.5	6.18E+08	9.78E+12	en-US	1216	1618	140	########	Houghton Mifflin Harcourt
36	The Lord of the Rings: Wea	Chris Smi	4.53	6.18E+08	9.78E+12	eng	218	19822	46	########	Houghton Mifflin Harcourt

11124 -header row=11123

Average: 21310.85696 Count: 11124 Sum: 237040662

Fig 53: calculate_statistics - Test Case

Input: The user needs to choose the option to calculate the statistical value for the input file.

Expected Output: Print out the statistical data correctly matching the actual values, which we can get by using the Excel sheet or others.

Conclusion

In conclusion, the Python-built Book Management System is an easy-to-use solution for effectively managing book data in a library. The program's modular design makes the code easy to read and maintain. It uses basic Python libraries like datetime and CSV, following design principles, including a single global variable and minimal external library use.

The system's functionalities, such as loading, saving, sorting, and searching books, allow users to interact seamlessly with their books in the library. It meets the outlined specifications and limitations, offering a valuable solution for book data management. Moreover, the system ensures reliable data processing, allowing users to manage their book collections, make modifications, and obtain statistical insights through a straightforward interface.

Responsibilities of each group member:

Report and Call Diagram - May Thu Kyaw, Mihini Methasa, Rattanan Suksen

Code:

1. List of books (Win Thawdar Aung + Rattanan Suksen)
2. Load (Rattanan Suksen)
3. Delete (May Thu Kyaw)
4. Sort (Mihini Methasa)
5. Search (May Thu Kyaw)
6. Save (Rattanan Suksen)
7. Insert a book (Rattanan Suksen)
8. Statistics (Win Thawdar Aung)
9. Info of a book (Mihini Methasa)
10. Update (Win Thawdar Aung)

Organizing the whole code - Win Thawdar Aung