

Assignment 7

Student ID: 901142

Student Name: Roshan Shrestha

Source code:

```
1 def main():
2     """
3     The main function that runs the library management system.
4     """
5     '''Create instance of the ModelLibrary class'''
6     lib = ModelLibrary()
7
8     # Execute the infinite loop until the user decides to quit
9     while True:
10        print('* * * * * Welcome to Library Management System * * * * *\nWhat would you like to do?\n1. Add new book\n2. Remove a book\n3. Search book by author\n4. Exit')
11        '''Ask user's input for desired action'''
12        option = input("Enter your choice: ")
13
14        # Perform the action based on user's input
15        if option == '1':
16            '''If input is 1, perform addition of book'''
17            title = input("Enter title of the book: ")
18            author = input("Enter the author of the book: ")
19            id = int(input("Enter the ID of the book: "))
20            lib.add_new_book(title, author, id)
21        elif option == '2':
22            '''If input is 2, remove the book'''
23            id = int(input("Enter the ID of the book to be removed: "))
24            lib.remove_book(id)
25        elif option == '3':
26            '''If input is 3, search the book'''
27            author = input("Enter the name of author: ")
28            books = lib.search_book_by_author(author)
29            if books:
30                print(f"The available books by {author} are:")
31                for book in books:
32                    print(book)
33        elif option == '4':
34            '''If input is 4, quit and exit the program'''
35            print("Thank you !")
36            break
37        else:
38            print("OOPS, invalid option selected, try again !")
39
40
41 if __name__ == "__main__":
42     main()
```

Figure 1: Source code for main function and performed actions.

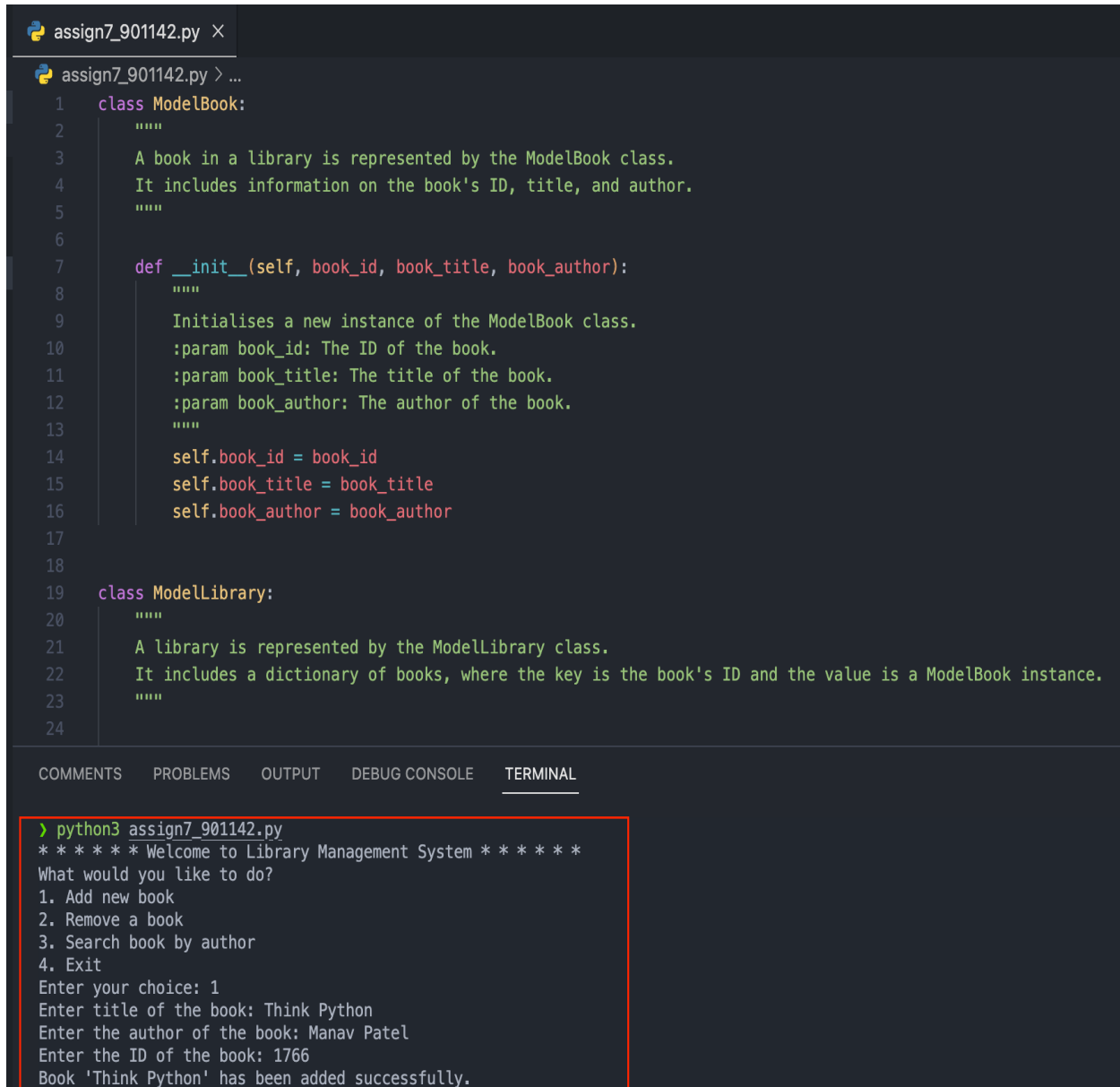
```

1  class ModelBook:
2      """
3      A book in a library is represented by the ModelBook class.
4      It includes information on the book's ID, title, and author.
5      """
6
7      def __init__(self, book_id, book_title, book_author):
8          """
9          Initialises a new instance of the ModelBook class.
10         :param book_id: The ID of the book.
11         :param book_title: The title of the book.
12         :param book_author: The author of the book.
13         """
14         self.book_id = book_id
15         self.book_title = book_title
16         self.book_author = book_author
17
18
19  class ModelLibrary:
20      """
21      A library is represented by the ModelLibrary class.
22      It includes a dictionary of books, where the key is the book's ID and the value is a ModelBook instance.
23      """
24
25      def __init__(self):
26          """
27          Initializes a new instance of the ModelLibrary class.
28          """
29          self.book_list = {}
30
31      def add_new_book(self, book_title, book_author, book_id):
32          """
33          Adds a new book to the library.
34          :param book_title: The title of the new book.
35          :param book_author: The author of the new book.
36          :param book_id: The ID of the new book.
37          """
38          if book_id in self.book_list:
39              print(f"Book ID {book_id} already exists. Cannot add book.")
40              return
41
42          new_book = ModelBook(book_id, book_title, book_author)
43          self.book_list[book_id] = new_book
44          print(f"Book '{book_title}' has been added successfully.")
45
46      def remove_book(self, book_id):
47          """
48          Removes a book from the library.
49          :param book_id: The ID of the book to be removed.
50          """
51          if book_id not in self.book_list:
52              print(f"No book found with ID {book_id}.")
53              return
54
55          removed_book = self.book_list.pop(book_id)
56          print(
57              f"Book '{removed_book.book_title}' has been removed successfully.")
58
59      def search_book_by_author(self, author_name):
60          """
61          Searches for books by the given author.
62          :param author_name: The name of the author to search for.
63          :return: A list of titles of books by the given author.
64          """
65          books_by_author = [
66              book.book_title for book in self.book_list.values() if book.book_author == author_name]
67
68          if not books_by_author:
69              print(f"No books found by author {author_name}.")
70              return []
71
72          return books_by_author

```

Figure 2: Source code for Classes for book and Library with function to add, remove and search book.

Test case for adding book:



The image shows a code editor with a file named `assign7_901142.py`. The code defines two classes: `ModelBook` and `ModelLibrary`. `ModelBook` has a docstring describing it as a class representing a book with attributes `book_id`, `book_title`, and `book_author`. It includes an `__init__` method that initializes these attributes. `ModelLibrary` has a docstring describing it as a class representing a library with a dictionary of books. Below the code editor, a terminal window shows the execution of the program. The terminal output displays a welcome message, a menu of options (1. Add new book, 2. Remove a book, 3. Search book by author, 4. Exit), and the user's input to add a new book. The user enters the title 'Think Python', the author 'Manav Patel', and the ID '1766'. The program then outputs 'Book 'Think Python' has been added successfully.'

```
assign7_901142.py > ...
1 class ModelBook:
2     """
3     A book in a library is represented by the ModelBook class.
4     It includes information on the book's ID, title, and author.
5     """
6
7     def __init__(self, book_id, book_title, book_author):
8         """
9         Initialises a new instance of the ModelBook class.
10        :param book_id: The ID of the book.
11        :param book_title: The title of the book.
12        :param book_author: The author of the book.
13        """
14        self.book_id = book_id
15        self.book_title = book_title
16        self.book_author = book_author
17
18
19 class ModelLibrary:
20     """
21     A library is represented by the ModelLibrary class.
22     It includes a dictionary of books, where the key is the book's ID and the value is a ModelBook instance.
23     """
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
> python3 assign7_901142.py
***** Welcome to Library Management System *****
What would you like to do?
1. Add new book
2. Remove a book
3. Search book by author
4. Exit
Enter your choice: 1
Enter title of the book: Think Python
Enter the author of the book: Manav Patel
Enter the ID of the book: 1766
Book 'Think Python' has been added successfully.
```

Figure 3: Adding a book.

Test case for searching book:

```
assign7_901142.py ×
assign7_901142.py > ...
1  class ModelBook:
2      """
3      A book in a library is represented by the ModelBook class.
4      It includes information on the book's ID, title, and author.
5      """
6
7      def __init__(self, book_id, book_title, book_author):
8          """
9          Initialises a new instance of the ModelBook class.
10         :param book_id: The ID of the book.
11         :param book_title: The title of the book.
12         :param book_author: The author of the book.
13         """
14         self.book_id = book_id
15         self.book_title = book_title
16         self.book_author = book_author
17
18
19  class ModelLibrary:
20      """
21      A library is represented by the ModelLibrary class.
22
23  COMMENTS  PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
> python3 assign7_901142.py
***** Welcome to Library Management System *****
What would you like to do?
1. Add new book
2. Remove a book
3. Search book by author
4. Exit
Enter your choice: 1
Enter title of the book: Think Python
Enter the author of the book: Manav Patel
Enter the ID of the book: 1766
Book 'Think Python' has been added successfully.
***** Welcome to Library Management System *****
What would you like to do?
1. Add new book
2. Remove a book
3. Search book by author
4. Exit
Enter your choice: 3
Enter the name of author: Manav Patel
The available books by Manav Patel are:
Think Python
***** Welcome to Library Management System *****
```

Figure 4: Search for book.

Test case for removing book:



The image shows a code editor window with a file named `assign7_901142.py`. The code defines two classes: `ModelBook` and `ModelLibrary`. `ModelBook` has a docstring describing it as a class representing a book with ID, title, and author, and an `__init__` method that initializes these attributes. `ModelLibrary` has a docstring stating it represents a library. Below the code editor is a terminal window with tabs for COMMENTS, PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. The terminal shows the execution of `python3 assign7_901142.py`, which starts a 'Welcome to Library Management System' menu. The user selects option 1 to add a book, enters 'Think Python' as the title, 'Manav Patel' as the author, and '1766' as the ID. The system confirms the book was added. Then, the user selects option 2 to remove a book, enters '1766' as the ID, and the system confirms the book 'Think Python' has been removed successfully. This final step is highlighted with a red rectangle.

```
assign7_901142.py ×
assign7_901142.py > ...
1  class ModelBook:
2      """
3      A book in a library is represented by the ModelBook class.
4      It includes information on the book's ID, title, and author.
5      """
6
7      def __init__(self, book_id, book_title, book_author):
8          """
9          Initialises a new instance of the ModelBook class.
10         :param book_id: The ID of the book.
11         :param book_title: The title of the book.
12         :param book_author: The author of the book.
13         """
14         self.book_id = book_id
15         self.book_title = book_title
16         self.book_author = book_author
17
18
19  class ModelLibrary:
20      """
21      A library is represented by the ModelLibrary class.
```

COMMENTS PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
> python3 assign7_901142.py
***** Welcome to Library Management System *****
What would you like to do?
1. Add new book
2. Remove a book
3. Search book by author
4. Exit
Enter your choice: 1
Enter title of the book: Think Python
Enter the author of the book: Manav Patel
Enter the ID of the book: 1766
Book 'Think Python' has been added successfully.
***** Welcome to Library Management System *****
What would you like to do?
1. Add new book
2. Remove a book
3. Search book by author
4. Exit
Enter your choice: 2
Enter the ID of the book to be removed: 1766
Book 'Think Python' has been removed successfully.
```

Figure 5: Removing a book.