**Library Management System Using Arrays**

**Problem Statement**

You are tasked with creating a library management system where users can perform multiple operations through a menu-driven interface. The system should use **arrays** to manage the collection of books. The program should include a menu with various operations, looping until the user decides to exit. Incorporate array-related operations such as searching, sorting, and dynamic management of book data.

**Requirements**

1. **Menu Options**:
   * **1**: Add a Book
   * **2**: View All Books
   * **3**: Search for a Book
   * **4**: Borrow a Book
   * **5**: Return a Book
   * **6**: Sort Books Alphabetically
   * **7**: Exit
2. **Operations**:
   * **Add a Book**: Allow the user to add a book's title to the library. Prevent duplicates. Ensure the library doesn't exceed a fixed capacity.
   * **View All Books**: Display all books in the library, along with their status (Available/Borrowed).
   * **Search for a Book**: Enable the user to search for a book by title. Indicate whether the book exists in the library and its availability.
   * **Borrow a Book**: Mark a book as borrowed if it is available. Prevent borrowing if the book is already borrowed.
   * **Return a Book**: Mark a borrowed book as available. Indicate an error if the book was not borrowed.
   * **Sort Books Alphabetically**: Sort the array of book titles alphabetically while maintaining their corresponding availability status.
   * **Exit**: Exit the program gracefully.
3. **Constraints**:
   * Fixed array size of 100 books.
   * Use two parallel arrays:
     + String[] books to store book titles.
     + boolean[] availability to store the availability status of each book (true for available, false for borrowed).
   * Implement input validation and error handling for invalid choices or operations.

**Expected Input/Output**

**Input 1:**

Menu:

1. Add a Book

2. View All Books

3. Search for a Book

4. Borrow a Book

5. Return a Book

6. Sort Books Alphabetically

7. Exit

Choose an option: 1

Enter book title: The Alchemist

**Output 1:**

Book added successfully!

**Input 2:**

yaml

Choose an option: 1

Enter book title: 1984

**Output 2:**

Book added successfully!

**Input 3:**

Choose an option: 6

**Output 3:**

python

Books sorted alphabetically!

**Input 4:**

Choose an option: 2

**Output 4:**

Books in the library:

1. 1984 - Available

2. The Alchemist - Available

**Input 5:**

Choose an option: 3

Enter book title: 1984

**Output 5:**

The book "1984" is available.

**Input 6:**

Choose an option: 7

**Output 6:**

Thank you for using the Library Management System!

**Hints**

* Use a do-while loop to repeat the menu until the user exits.
* Use a switch statement for menu options.
* Use nested loops for searching and sorting operations.
* Validate inputs to prevent invalid operations (e.g., borrowing a non-existent book).