**Exercise 3: Implementing Logging with Spring AOP**

Scenario:

The library management application requires logging capabilities to track method execution times.

**1. Introduction**

This document details the implementation of logging capabilities in the Library Management System using Spring AOP. The logging feature is used to track method execution times, which is crucial for performance monitoring and debugging.

**2. Steps and Updates**

**2.1. Add Spring AOP Dependency**

The pom.xml file is updated to include the Spring AOP dependency, allowing the use of AOP features in the application.

**2.2. Create an Aspect for Logging**

A new package com.library.aspect is created, and a LoggingAspect class is added. This class includes a method logExecutionTime that logs the execution time of methods in the com.library.service package.

**Created File:** src/main/java/com/library/aspect/LoggingAspect.java

**2.3. Enable AspectJ Support**

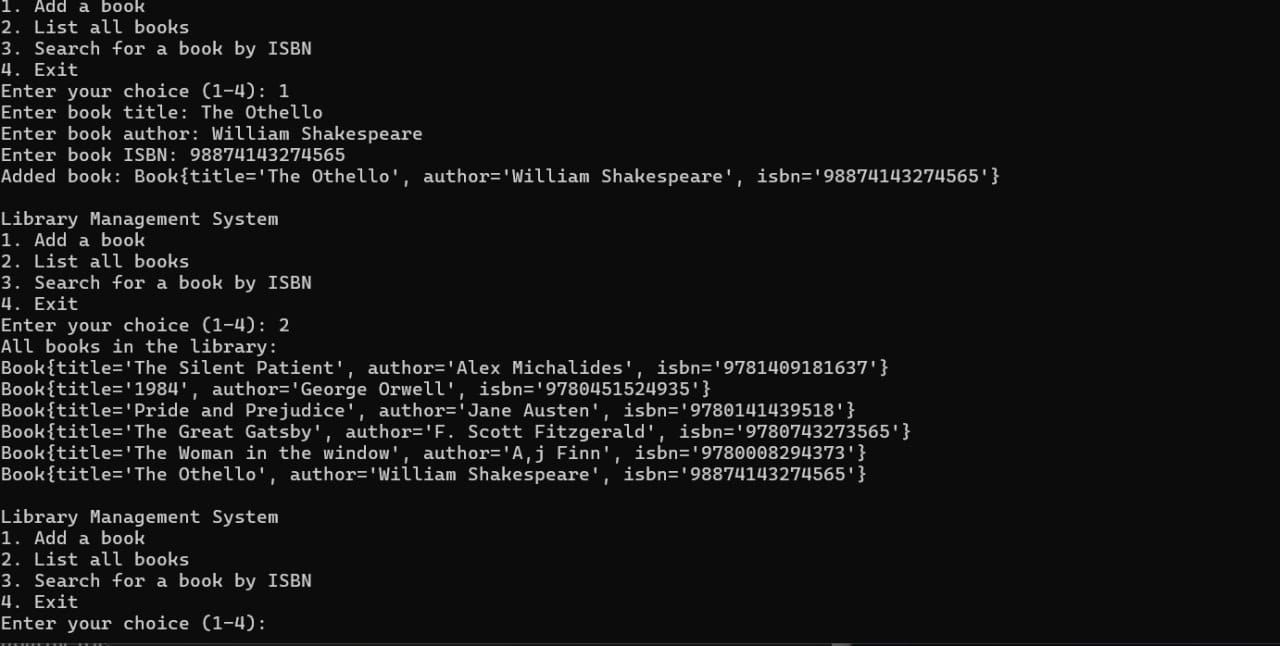
The applicationContext.xml is updated to enable AspectJ support and register the LoggingAspect bean.

**Updating the applicationContext.xml Section.**

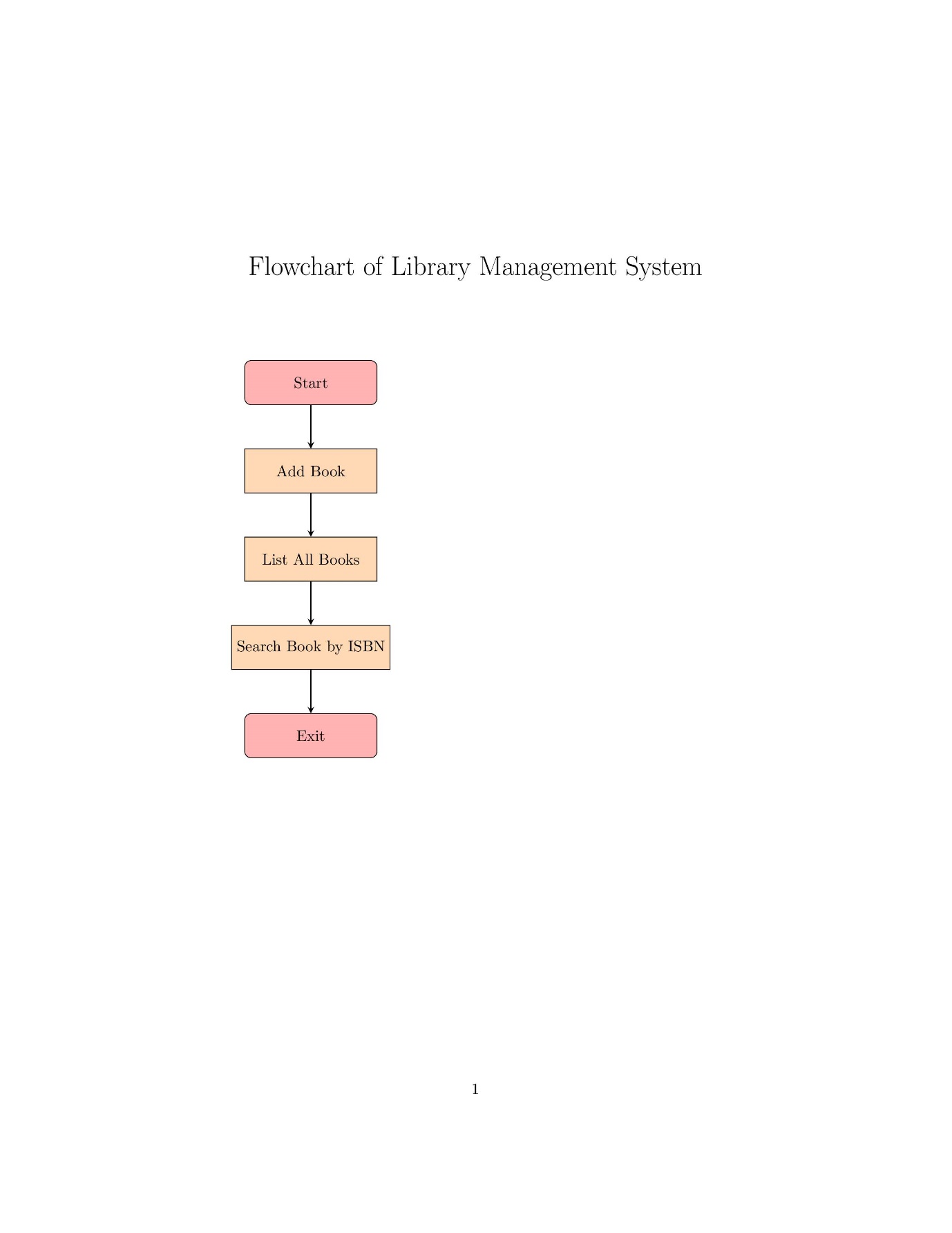
**2.4. Test the Aspect**

Running the LibraryManagementApplication class will now produce console output showing the execution time of methods in the BookService class, as handled by the LoggingAspect.

**OUTPUT :**



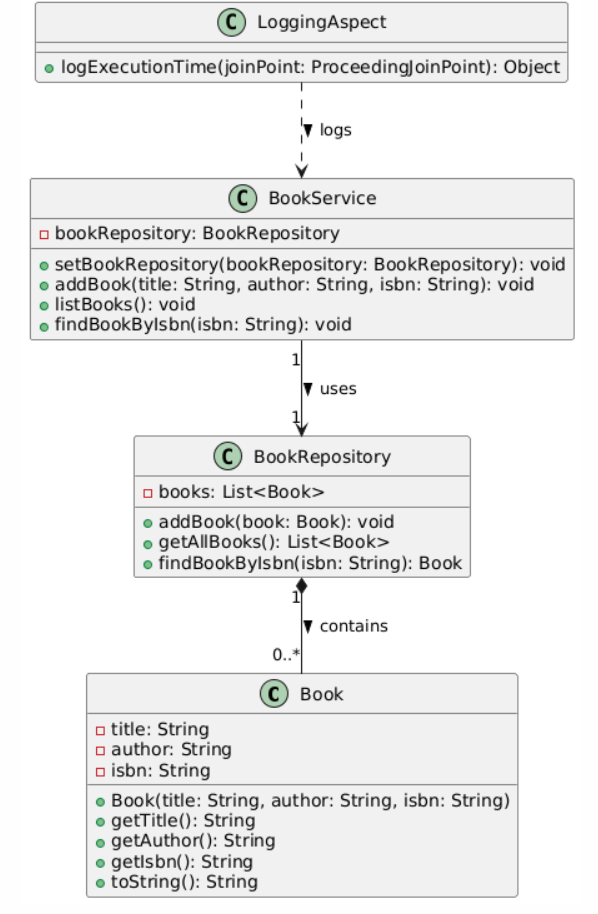
**FLOWCHART of the program :**



The flowchart represents the logical flow of the Library Management System. It starts with the main process, followed by user interactions like adding a book, listing all books, and searching for a book by ISBN. The aspect-oriented programming (AOP) feature is not directly depicted in the flowchart but is implicitly integrated into the process to log method execution times.

**Description:**

* **Start**: Initiates the application.
* **Add Book**: Adds a new book to the repository.
* **List All Books**: Retrieves and displays all books from the repository.
* **Search Book by ISBN**: Searches for a book using its ISBN.
* **Exit**: Ends the application.

**CLASS DIAGRAM :** 

The class diagram provides an overview of the classes in the Library Management System and their relationships. It shows how the BookService interacts with BookRepository and how the LoggingAspect class is related to the BookService.

**Key Components:**

* **Book**: Represents a book entity with attributes like title, author, and isbn.
* **BookRepository**: Manages the collection of books, providing methods to add and retrieve books.
* **BookService**: Acts as a service layer, interacting with the BookRepository to perform operations related to books.
* **LoggingAspect**: An aspect class that logs the execution time of methods in the BookService class.