BOOKSTORE API PROJECT REPORT  
  
**1. Introduction**

The Bookstore API is a Spring Boot-based RESTful web service that provides functionality for managing books and authors in a digital library system. It allows clients to perform CRUD operations on books and authors, supports filtering, pagination, and integrates with Swagger for API documentation.

**2. Abstract**

This project is designed to demonstrate the capabilities of Spring Boot in building a scalable and maintainable backend API. It includes features such as creating and retrieving books and author records, filtering based on title or author name, paginated responses, and integrated Swagger UI for testing endpoints. The database used is H2 for easy in-memory testing and development.

**3. Tools Used**

* **Backend Framework**: Spring Boot (v3.2.5)
* **Database**: H2 (File-based persistence)
* **Build Tool**: Apache Maven
* **IDE**: Eclipse
* **API Testing**: Postman
* **API Documentation**: Springdoc OpenAPI (Swagger)
* **Language**: Java

**4. Steps Involved in Building the Project**

1. **Project Initialization**: Created a Maven project with Spring Boot dependencies (Web, JPA, H2, Swagger).
2. **Entity Creation**: Defined Book and Author entities with appropriate relationships (many-to-one).
3. **Repository Layer**: Implemented JpaRepository interfaces for database operations.
4. **Service Layer**: Developed business logic for CRUD operations, pagination, and filtering using JPA Criteria API.
5. **Controller Layer**: Created RESTful endpoints for books and authors, with support for sorting and search.
6. **Database Configuration**: Configured H2 database for persistence and enabled H2 console.
7. **Swagger Setup**: Integrated Swagger using springdoc-openapi for API testing and visualization.
8. **Testing**: Used Postman to test all endpoints for proper data creation, retrieval, and filtering.
9. **Error Handling**: Added basic exception handling for resource not found and validation.

**5. Conclusion**

The Bookstore API demonstrates how to build a robust RESTful service using modern Spring Boot practices. It showcases the use of layered architecture, pagination, Swagger documentation, and H2 for testing. This project can serve as a foundational backend for larger library or inventory systems and can easily be extended with features like authentication, validation, and deployment to production servers.