**Exercise 10: Online Bookstore - Configuring Content Negotiation**

**BookController**

package com.bookstore.api.controllers;

import com.bookstore.api.dtos.BookDTO;

import com.bookstore.api.entities.Book;

import com.bookstore.api.mappers.BookMapper;

import com.bookstore.api.services.BookService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.http.HttpHeaders;

import org.springframework.http.HttpStatus;

import org.springframework.http.ResponseEntity;

import org.springframework.web.bind.annotation.\*;

import javax.validation.Valid;

@RestController

@RequestMapping("/books")

public class BookController {

private final BookService bookService;

private final BookMapper bookMapper;

@Autowired

public BookController(BookService bookService, BookMapper bookMapper) {

this.bookService = bookService;

this.bookMapper = bookMapper;

}

@PostMapping

public ResponseEntity<BookDTO> createBook(@RequestBody @Valid BookDTO bookDTO, @RequestHeader("Accept") String acceptHeader) {

Book book = bookMapper.bookDTOToBook(bookDTO);

book = bookService.createBook(book);

if (acceptHeader.equals("application/json")) {

return new ResponseEntity<>(bookMapper.bookToBookDTO(book), HttpStatus.CREATED);

} else if (acceptHeader.equals("application/xml")) {

return new ResponseEntity<>(bookMapper.bookToBookXML(book), HttpStatus.CREATED);

} else {

return new ResponseEntity<>(HttpStatus.NOT\_ACCEPTABLE);

}

}

@GetMapping

public ResponseEntity<?> getAllBooks(@RequestHeader("Accept") String acceptHeader) {

List<Book> books = bookService.getAllBooks();

if (acceptHeader.equals("application/json")) {

return new ResponseEntity<>(books.stream().map(bookMapper::bookToBookDTO).toList(), HttpStatus.OK);

} else if (acceptHeader.equals("application/xml")) {

return new ResponseEntity<>(books.stream().map(bookMapper::bookToBookXML).toList(), HttpStatus.OK);

} else {

return new ResponseEntity<>(HttpStatus.NOT\_ACCEPTABLE);

}

}

// ...

}

In the above code, i have configured Spring Boot to support content negotiation.

- I have implemented logic to produce and consume different media types (JSON, XML) based on the Accept header.

- I have used the @RequestHeader annotation to inject the Accept header value into the controller methods.

- I have used if-else statements to determine the media type and return the appropriate response.

Note that I have assumed that the BookMapper class has methods to convert between Book and BookDTO (for JSON) and BookXML (for XML).