**Exercise 9: Creating a Spring Boot Application**

Book.java

package com.Librarymanagement.LibraryManagement.Model;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

@Entity

public class Book {

    @Id

    @GeneratedValue(strategy = GenerationType.IDENTITY)

    private Long id;

    private String title;

    private String author;

    public Long getId() { return id; }

    public void setId(Long id) { this.id = id; }

    public String getTitle() { return title; }

    public void setTitle(String title) { this.title = title; }

    public String getAuthor() { return author; }

    public void setAuthor(String author) { this.author = author; }

}

BookRepository.java

package com.Librarymanagement.LibraryManagement.repository;

import com.Librarymanagement.LibraryManagement.Model.Book;

import org.springframework.data.jpa.repository.JpaRepository;

public interface BookRepository extends JpaRepository<Book, Long> {

}

BookController.java

package com.Librarymanagement.LibraryManagement.controller;

import com.Librarymanagement.LibraryManagement.Model.Book;

import com.Librarymanagement.LibraryManagement.repository.BookRepository;

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

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

import java.util.List;

import java.util.Optional;

@RestController

@RequestMapping("/api/books")

public class BookController {

    @Autowired

    private BookRepository bookRepository;

    @GetMapping

    public List<Book> getAllBooks() {

        return bookRepository.findAll();

    }

    @GetMapping("/{id}")

    public Optional<Book> getBookById(@PathVariable Long id) {

        return bookRepository.findById(id);

    }

    @PostMapping

    public Book createBook(@RequestBody Book book) {

        return bookRepository.save(book);

    }

    @PutMapping("/{id}")

    public Book updateBook(@PathVariable Long id, @RequestBody Book bookDetails) {

        return bookRepository.save(bookDetails);

    }

    @DeleteMapping("/{id}")

    public void deleteBook(@PathVariable Long id) {

        bookRepository.deleteById(id);

    }

}

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.