**Assignment 13**

**Name: Rahul Kumar**

**Roll No: 13000116080**

**Problem Statement:** Extend your solution under LA12 to cover the Book maintenance and Student Maintenance functions. Maintenance typically includes addition, modification, deletion and query functions. As for addition, your system should allow single item entry by the user or bulk entries from a file.

**Code**

package darkarmy.springframework.librarymanagementsystem.model;

import lombok.AllArgsConstructor;

import lombok.Data;

import lombok.NoArgsConstructor;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class Book {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Integer id;

private String title;

private String author;

private String price;

private String quantity;

private boolean isAvailable;

public Book() {

}

public Book(Integer id, String title, String author, String price, String quantity, boolean isAvailable) {

this.id = id;

this.title = title;

this.author = author;

this.price = price;

this.quantity = quantity;

this.isAvailable = isAvailable;

}

public Integer getId() {

return id;

}

public void setId(Integer 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;

}

public String getPrice() {

return price;

}

public void setPrice(String price) {

this.price = price;

}

public String getQuantity() {

return quantity;

}

public void setQuantity(String quantity) {

this.quantity = quantity;

}

public boolean isAvailable() {

return isAvailable;

}

public void setAvailable(boolean available) {

isAvailable = available;

}

@Override

public String toString() {

return "Book{" +

"id=" + id +

", title='" + title + '\'' +

", author='" + author + '\'' +

", price='" + price + '\'' +

", quantity='" + quantity + '\'' +

", isAvailable=" + isAvailable +

'}';

}

}

package darkarmy.springframework.librarymanagementsystem.controller;

import darkarmy.springframework.librarymanagementsystem.model.Book;

import darkarmy.springframework.librarymanagementsystem.repository.BookRepository;

import org.springframework.http.HttpStatus;

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

import java.util.List;

import java.util.Optional;

@RestController

@RequestMapping("/book")

public class BookController {

private BookRepository bookRepository;

public BookController(BookRepository bookRepository) {

this.bookRepository = bookRepository;

}

@GetMapping("/all")

@ResponseStatus(HttpStatus.OK)

public List<Book> getAllBooks(){

return bookRepository.findAll();

}

@GetMapping("/{id}")

@ResponseStatus(HttpStatus.OK)

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

return bookRepository.findById(id);

}

@PostMapping("/add")

@ResponseStatus(HttpStatus.CREATED)

public Book addBook(@RequestBody Book book){

return bookRepository.save(book);

}

@PostMapping("/update/{id}")

@ResponseStatus(HttpStatus.OK)

public String updateBook(@PathVariable Integer id, @RequestBody Book book){

Optional<Book> book1 = bookRepository.findById(id);

if(book1!=null){

bookRepository.deleteById(id);

bookRepository.save(book);

return "Book "+ book.getTitle()+" updated successfully! "+ book.toString();

}else{

return "Invalid book id";

}

}

@DeleteMapping("/{id}")

@ResponseStatus(HttpStatus.OK)

public String deleteBook(@PathVariable Integer id){

bookRepository.deleteById(id);

return "Book with id "+id+" deleted succesfully!";

}

}

**Output**

