**PROBLEM STATEMENT – 3**

**Create View layer with Spring Boot Thymeleaf**

**pom.xml:**

<dependency>

    <groupId>org.springframework.boot</groupId>

    <artifactId>spring-boot-starter-thymeleaf</artifactId>

</dependency>

**books.html:**

<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

<head>

    <title>Book List</title>

</head>

<body>

    <h1>Book List</h1>

    <table>

        <thead>

            <tr>

                <th>ID</th>

                <th>Title</th>

                <th>Publisher</th>

                <th>ISBN</th>

                <th>Pages</th>

                <th>Year</th>

                <th>Actions</th>

            </tr>

        </thead>

        <tbody>

            <tr th:each="book : ${books}">

                <td th:text="${book.id}"></td>

                <td th:text="${book.bookTitle}"></td>

                <td th:text="${book.bookPublisher}"></td>

                <td th:text="${book.bookIsbn}"></td>

                <td th:text="${book.bookNumberOfPages}"></td>

                <td th:text="${book.bookYear}"></td>

                <td>

                    <a th:href="@{/books/edit/{id}(id=${book.id})}">Edit</a>

                    <a th:href="@{/books/delete/{id}(id=${book.id})}">Delete</a>

                </td>

            </tr>

        </tbody>

    </table>

    <a href="/books/add">Add New Book</a>

</body>

</html>

**add-book.html:**

<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

<head>

    <title>Add Book</title>

</head>

<body>

    <h1>Add Book</h1>

    <form action="#" th:action="@{/books}" th:object="${book}" method="post">

        <label>Title: <input type="text" th:field="\*{bookTitle}" /></label><br/>

        <label>Publisher: <input type="text" th:field="\*{bookPublisher}" /></label><br/>

        <label>ISBN: <input type="text" th:field="\*{bookIsbn}" /></label><br/>

        <label>Pages: <input type="number" th:field="\*{bookNumberOfPages}" /></label><br/>

        <label>Year: <input type="number" th:field="\*{bookYear}" /></label><br/>

        <button type="submit">Add Book</button>

    </form>

    <a href="/books">Back to Book List</a>

</body>

</html>

**edit-book.html:**

<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

<head>

    <title>Edit Book</title>

</head>

<body>

    <h1>Edit Book</h1>

    <form action="#" th:action="@{/books}" th:object="${book}" method="post">

        <input type="hidden" th:field="\*{id}" />

        <label>Title: <input type="text" th:field="\*{bookTitle}" /></label><br/>

        <label>Publisher: <input type="text" th:field="\*{bookPublisher}" /></label><br/>

        <label>ISBN: <input type="text" th:field="\*{bookIsbn}" /></label><br/>

        <label>Pages: <input type="number" th:field="\*{bookNumberOfPages}" /></label><br/>

        <label>Year: <input type="number" th:field="\*{bookYear}" /></label><br/>

        <button type="submit">Update Book</button>

    </form>

    <a href="/books">Back to Book List</a>

</body>

</html>

**BookStoreRestController.java:**

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

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

import java.util.List;

@Controller

@RequestMapping("/books")

public class BookStoreRestController {

    @Autowired

    private BookStoreService service;

    @GetMapping

    public String getAllBooks(Model model) {

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

        model.addAttribute("books", books);

        return "books";

    }

    @GetMapping("/add")

    public String showAddBookForm(Model model) {

        model.addAttribute("book", new Book());

        return "add-book";

    }

    @PostMapping

    public String addBook(@ModelAttribute Book book) {

        service.addBook(book);

        return "redirect:/books";

    }

    @GetMapping("/edit/{id}")

    public String showEditBookForm(@PathVariable Long id, Model model) {

        Book book = service.getBookById(id);

       .addAttribute("book", book);

        return "edit-book";

    }

    @PostMapping("/edit")

    public String updateBook(@ModelAttribute Book book) {

        service.updateBook(book);

        return "redirect:/books";

    }

    @GetMapping("/delete/{id}")

    public String deleteBook(@PathVariable Long id) {

        service.deleteBookById(id);

        return "redirect:/books";

    }

}