

Problem Set 5

The following problem set will be worth 100 points. The code will be submitted electronically via Canvas using the “Problem Set 5” dropbox. The assignment is **due at the start of the class two weeks from the date it was assigned**.

Your code will be graded on both elegance and *user-friendliness*.

Exercise #1 –Text Editor Problem (100pts)

Overview

During college, you work at your local bookstore named “Moby Dickens” to help supplement your income. It’s a small “mom and pop” store, and doesn’t have an online web presence as other national bookstores do. The owners of the store realize that you are studying Java and web development, and came to you with a request, to build a website for their book store. They also want users to be able to view their current inventory, and administrative users to be able to add/remove inventory from their database.

Requirements

For this problem set, you will build a Spring Boot application which when ran will demo their web application. At a core, a **Book** can be thought of as an **Object** in Java as the following properties:

- Book ID (Integer)
- Title (String)
- ISBN (Long)
- Author (String)
- Date Published (DateTime)
- Genre (Genre)
- Price (Double)

A **Genre** can be represented as an enum with a variety of values (horror, non-fiction, romance, classics.)

Since we haven’t discussed the concept of a database yet in class, you can use a **List<Book>** behind the scenes to represent your database. Your web application should also feature security, where the user named **admin** with password **123456** is able to add/edit/delete records from the database.

Page Requirements

To receive full credit for this assignment, your web application must have the following URLs at a minimum:

- **/**
 - The home page for Moby Dickens
- **/search**
 - A public accessible search page where users can view books by searching for Name and Author
- **/contact-us**
 - A web form where users can email questions to (admin@mobydickens.com), capturing the requester name, email address and also a subject and body for the message.
- **/login**
 - The admin given above can log into the back end portal for the web site

The following pages should also be secured and only accessible by the administrative user:

- **/admin/books/view**
 - A page which lists all books in the database
- **/admin/books/add**
 - A page where an administrator can add a book to the database
- **/admin/books/edit/{bookId}**
 - A page where the administrator can edit the details of a particular bookstore
- **/admin/books/delete/{bookId}**
 - A page where the administrator can delete a book from the database
- **/logout**
 - The logout page which logs out the user and returns the user back to the home page

Points will be awarded based on creativity and correctness of the assignment. Extra credit could be awarded for students who go “above and beyond” the requirements of the assignment. More details will be given regarding more detailed parts of the assignment (for instance submitting email in code.)

Submit all files (including the Gradle script I can use to run your application) in a zip file.

Submission Requirements: Submit the aforementioned files in a zip file with the naming strategy:

First initial + last name + PS + problem set number.zip

As an example, I would submit the code in a zip file named **mmeluskyPS5.zip**. Submit your zip file via the “Problem Set 4” Canvas dropbox before the date of the close of the assignment.