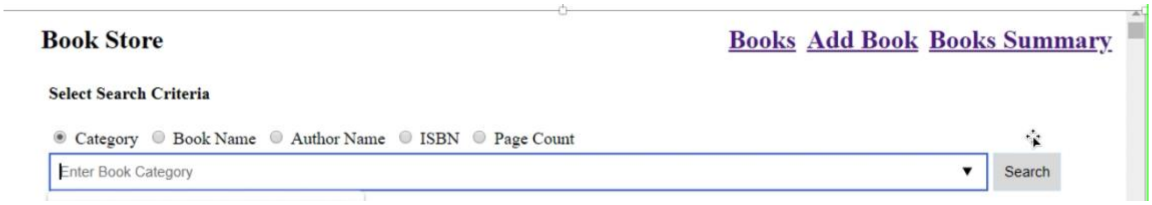
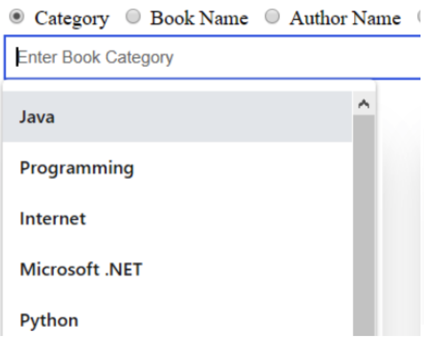
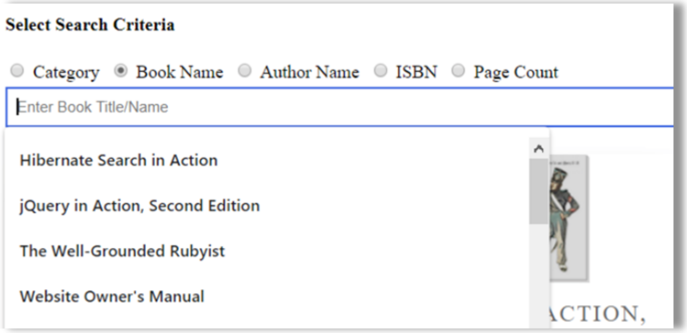
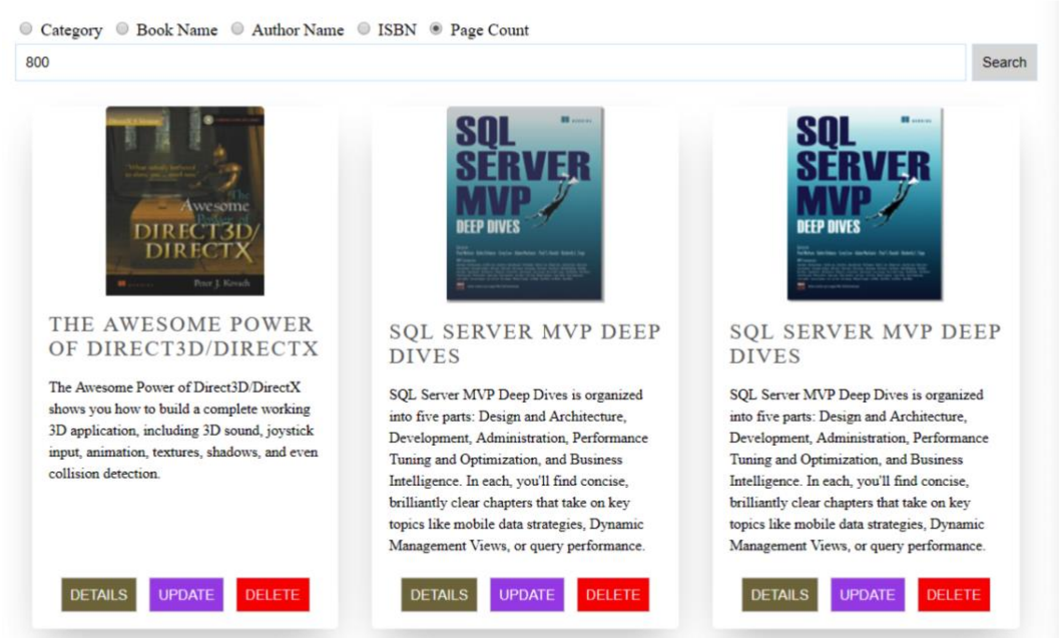
Assessment 7

Extend the Book Store to make the app functionality accessible via Web UI.

Deadline – Next week 1 hour before the Lab

1. Open ***Lab9- Assessments /BookStore*** in WebStorm. Run **npm install** to install the packages.
2. Design and implement a Web U I to make the App functionality accessible to users:

Web Page	**Functionality**
index.html	First, the index page should have a main navigation menu providing two links: - *Books* link is the home page that allows searching for books. - *Add Book* link allows adding a book. -BookSummary The index page should allow the user to enter the parameters to search for books by *category*, *name*, *isbn*, *author* or *pageCount*.  Book Store Books Add Book Books Summary Select Search Criteria ● Category ○ Book Name ○ Author Name ○ ISBN ○ Page Count Enter Book Category Search The Book category and the Author parameter should be a **dropdown** filled with the data returned from /api/categories and /api/authors respectively.  ● Category ○ Book Name ○ Author Name Enter Book Category Java Programming Internet Microsoft .NET Python  Select Search Criteria ○ Category ● Book Name ○ Author Name ○ ISBN ○ Page Count Enter Book Title/Name Hibernate Search in Action jQuery in Action, Second Edition The Well-Grounded Rubyist Website Owner's Manual Upon submission of the search parameters, the app should fetch the books then display them in a table. Based on the selected search option the app should make use of the Web API developed in Lab 7 to fetch the books. You should use the Web API implementation in the provided base solution. Besides each book displayed in the table, the app should offer the following links: - details: this link should display the book details page. - update: this link should allow updating the book. - delete: this link should allow deleting the book. After the user confirmation, the app should delete the book using the Web API @ /api/books/:isbn =>**See image below**



○ Category ○ Book Name ○ Author Name ○ ISBN ● Page Count

800

THE AWESOME POWER OF DIRECT3D/DIRECTX

The Awesome Power of Direct3D/DirectX shows you how to build a complete working 3D application, including 3D sound, joystick input, animation, textures, shadows, and even collision detection.

SQL SERVER MVP DEEP DIVES

SQL Server MVP Deep Dives is organized into five parts: Design and Architecture, Development, Administration, Performance Tuning and Optimization, and Business Intelligence. In each, you'll find concise, brilliantly clear chapters that take on key topics like mobile data strategies, Dynamic Management Views, or query performance.

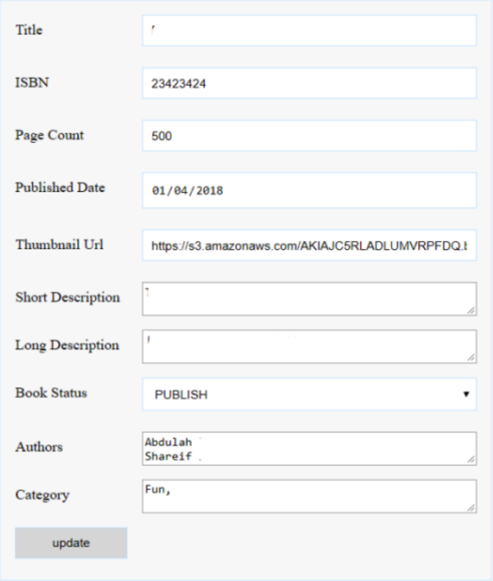
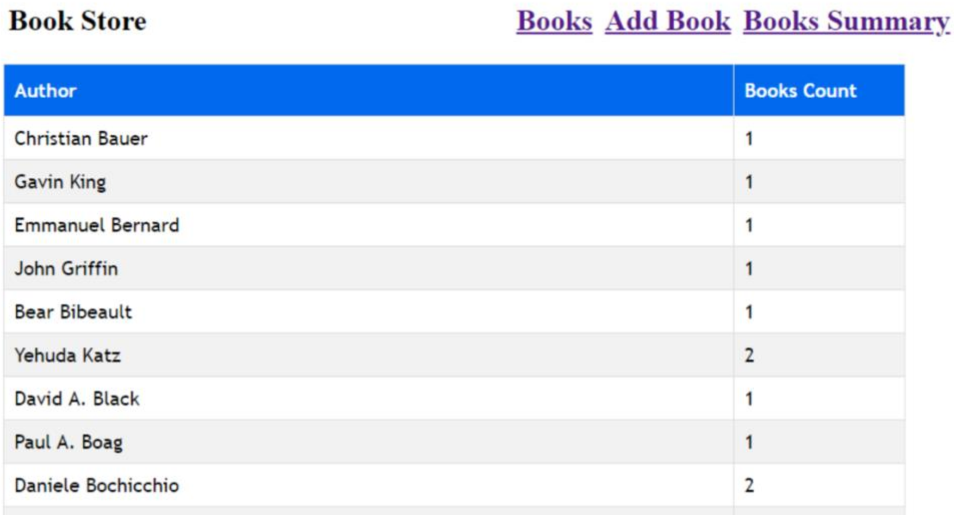
SQL SERVER MVP DEEP DIVES

SQL Server MVP Deep Dives is organized into five parts: Design and Architecture, Development, Administration, Performance Tuning and Optimization, and Business Intelligence. In each, you'll find concise, brilliantly clear chapters that take on key topics like mobile data strategies, Dynamic Management Views, or query performance.

Reminder of books API from Lab 7:

Get	*/api/books/:isbn*	Gets a book by isbn.
Get	/api/books?category=	Returns all the books for a particular category. If the **category** is not provided then you get all the books otherwise you should get the books that match that category. E.g. *category = Programming* should get all the programming books.
Get	/api/books?name=	Get the books where the name matches the name parameter.
Get	/api/books?pageCount=	Get the books with pages >= the pageCount parameter. E.g. if the pageCount=200 you should get all the books with pages >= 200.
Get	/api/books?author=	Display all the books authored by that specific author.

book.html	This page displays the details of the selected book. The book details should be fetched using */api/books/:isbn*

book-editor.html	Provides a form to allow adding a book to the book-catalog. The entered book should be posted to the Web API @ /api/books/ The form can also be used for updating an existing book. The updated book should be sent to the Web API @ /api/books/ **See image below**  Title: ISBN: 23423424 Page Count: 500 Published Date: 01/04/2018 Thumbnail Url: https://s3.amazonaws.com/AKIAJC5RLADLUMVRPFDQ.t Short Description: Long Description: Book Status: PUBLISH Authors: Abdulah Shareif Category: Fun,
books-summary.html	Display a table that contains the author name and the number of books they have authored. The data for this page should be fetched using *Get /api/books/summary*  Author – Books Count Christian Bauer – 1 Gavin King – 1 Emmanuel Bernard – 1 John Griffin – 1 Bear Bibeault – 1 Yehuda Katz – 2 David A. Black – 1 Paul A. Boag – 1 Daniele Bochicchio – 2

Make sure the app main menu is accessible in all the app pages.

3. Test the **BookStore** Web UI and provide screenshots as evidence in the testing document.

   After you complete the lab, fill in the ***TestingDoc-Grading-Sheet.docx*** and submit your work under **Assessment7-Web UI** folder. Sync your repository to push your work to Github.