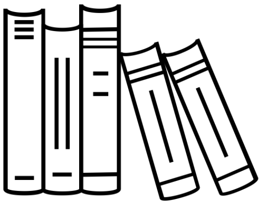
Book Shop - Part II

Overview

In this part of the exercise we will add new functionality to the Book Shop app. This will include



- More advanced filtering of the book list
- Sorting the book list
- Pagination
- Using modals for user input
- Making the state of our app bookmarkable

The Model

1. First, let's add a **rating** property to our model. This will be a number between 1 and 5. Our model will now look like this:

```
{
  id: bg4J78,
  title: 'The adventures of Lori Ipsi',
  price: 120,
  imgUrl: 'lori-ipsi.jpg',
  rating: 3
}
```

2. Delete the books from the browser's local storage and refactor the service to produce demo data which includes random book ratings.

3. Update the book list to display book ratings. ⇩

4. Update the book details modal to display the book's rating. ⇩

Filtering, Sorting and Pagination

1. Add support for combined filtering of the book list by **title** and **rating** (minimum rating). ⇩

   - Use a `filterBy` object as demonstrated in class
   - Use a dropdown to choose the minimum rating
   - Update the clear filter button to clear both criteria

2. Reflect the filter values in the ***query string params***. (Make the book list filter state book-markable) ⇩
3. Add support for sorting the book list by title, price or rating. ⇩

   - Use a `sortBy` object as demonstrated in class
   - Add a two radio buttons to choose between ascending or descending sort order

4. Add support for pagination - show 5 books on every page and add **next** and **prev** buttons to navigate between pages. ⇩
5. Make the page navigation cyclic - when the **next** button is clicked on the last page, it will navigate back to the first page and when **prev** is clicked on the first page it will navigate to the last page. ⇩

Using Modals to Improve the UX

Let’s improve our user experience by implementing modals for adding new books and editing existing book details .

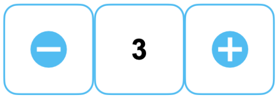
1. **Adding a book**
   Refactor ***`onAddBook()`*** to use a modal - we will call it the “book edit” modal. ⇩

2. **Updating a book**
   Refactor ***`onUpdateBook()`*** to display a modal which will let the user edit the book details and save them, using an ***`updateBook()`*** function from the service. Can you reuse the same modal you created for the add book operation? ⇩

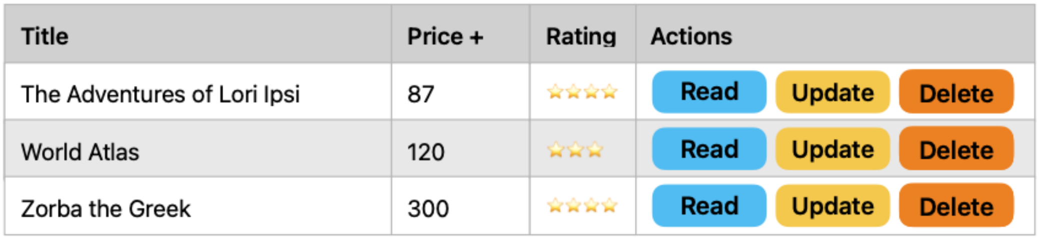
Bonus Features

UI / UX Features

1. Allow editing the book rating in the book edit modal, using **+** / **–** buttons. Something like this: ⇩



2. Show the book rating in the book list using stars. (see the image below) ⇩
3. Make the sorting functionality work by clicking the table column headers. Show a **+** or **–** sign next to the column heading by which the list is sorted indicating the sort direction. In the example below, the list is sorted by book price in ascending order ⇩



Title	Price +	Rating	Actions
The Adventures of Lori Ipsi	87	★★★★	Read Update Delete
World Atlas	120	★★★	Read Update Delete
Zorba the Greek	300	★★★★	Read Update Delete

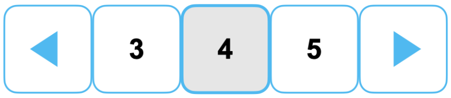
Other Features

1. Add book-markability to the sort & pagination state so that the app can be loaded with the book list sorted and in the specific page which was bookmarked. ⇩

2. Add book-markability to the Book Details modal state so that the app can be loaded with the full details of a specific book displayed. ⇩

3. Add a button to the book details modal which will navigate to the book edit modal of the same book. ⇩

4. Support more advanced pagination which allows navigating directly to a page by its number and shows which page is currently displayed. ⇩
   Something like this:



Happy Coding :-)