## **PROJECT: LIBRARY**

# Introduction

Let’s extend the ‘Book’ example from the previous lesson and turn it into a small Library app.

## Assignment

1. If you haven’t already, set up your project with skeleton HTML/CSS and JS files.
2. All of your book objects are going to be stored in a simple array, so add a function to the script (not the constructor) that can take the user's input and store the new book objects into an array. Your code should look something like this:  
   let myLibrary = []; function Book() { // the constructor... } function addBookToLibrary() { // do stuff here }
3. Write a function that loops through the array and displays each book on the page. You can display them in some sort of table, or each on their own “card”. It might help for now to manually add a few books to your array so you can see the display.
4. Add a “NEW BOOK” button that brings up a form allowing users to input the details for the new book: author, title, number of pages, whether it’s been read and anything else you might want.
5. Add a button on each book’s display to remove the book from the library.
   1. You will need to associate your DOM elements with the actual book objects in some way. One easy solution is giving them a data-attribute that corresponds to the index of the library array.
6. Add a button on each book’s display to change its read status.
   1. To facilitate this you will want to create the function that toggles a book’s read status on your Book prototype instance.
7. Optional -we haven’t learned any techniques for actually storing our data anywhere, so when the user refreshes the page all of their books will disappear! If you want, you are capable of adding some persistence to this library app using one of the following techniques:
   1. localStorage ([docs here](https://developer.mozilla.org/en-US/docs/Web/API/Web_Storage_API/Using_the_Web_Storage_API)) allows you to save data on the user’s computer. The downside here is that the data is ONLY accessible on the computer that it was created on. Even so, it’s pretty handy! Set up a function that saves the whole library array to localStorage every time a new book is created, and another function that looks for that array in localStorage when your app is first loaded. (make sure your app doesn’t crash if the array isn’t there!)
   2. Firebase ([check it out!](https://firebase.google.com/docs/?authuser=0)) is an online database that can be set up relatively easily, allowing you to save your data to a server in the cloud! Teaching you how to use it is beyond the scope of this tutorial, but it is almost definitely within your skill set. If you’re interested, check out [this video](https://www.youtube.com/watch?v=noB98K6A0TY) to see what it’s all about.

## Examples

* [**https://igorashs.github.io/library/**](https://igorashs.github.io/library/)
* [**https://michalosman.github.io/library/**](https://michalosman.github.io/library/)
* [**https://zakarya-mks.github.io/my-library/**](https://zakarya-mks.github.io/my-library/)
* [**https://ginnerzapata.github.io/library-app/**](https://ginnerzapata.github.io/library-app/)