# JS 101

Your task is to build your own digital library using JavaScript. You will be focusing on building most of the interface components as a proof of concept. You will build your app using many important aspects of JavaScript including, objects, functions, primitives, loops, and use the console for debugging purposes.

## Requirements

#### Library

- Library code should utilize functions
- Your library must have a “books” property which must be an array.
    - The array will contain book objects
- Your library must have the following public functions

    - **addBook(book)**
      *Purpose:* Add a book object to your books array.
      *Return: boolean* true if it is not already added, false if it is already added.

    - **removeBookByTitle(title)**
      *Purpose:* Remove book from from the books array by its title.
      *Return: boolean* true if the book(s) were removed, false if no books match

    - **removeBookByAuthor(authorName)**
      *Purpose:* Remove a specific book from your books array by the author name.
      *Return: boolean* true if the book(s) were removed, false if no books match

    - **getRandomBook()**
      *Purpose:* Return a random book object from your books array
      *Return: book object* if you find a book, *null* if there are no books

    - **getBookByTitle(title)**
      *Purpose:* Return all books that completely or partially matches the string title passed into the function
      *Return: array of book objects* if you find books with matching titles, *empty array* if no books are found

    - **getBooksByAuthor**(**authorName**)
      *Purpose:* Finds all books where the author’s name partially or completely matches the authorName argument passed to the function.
      *Return: array of books* if you find books with match authors, *empty array* if no books match

- **addBooks(books)**
  *Purpose:* Takes multiple books, in the form of an array of book objects, and adds the objects to your books array.
  *Return: number* number of books successfully added, *0* if no books were added

- **getAuthors()**
  *Purpose:* Find the distinct authors' names from all books in your library
  *Return: array of strings* the names of all distinct authors, *empty array* if no books exist or if no authors exist

- **getRandomAuthorName()**
  *Purpose:* Retrieves a random author name from your books collection
  *Return: string* author name, *null* if no books exist

**Book**

- A book must be an object
- A book object must have the following properties:
  - **title** - s*tring* representing the title of the song
  - **author** - *string* representing the authors name
  - **numberOfPages** - *number* representing the total page numbers
  - **publishDate** - *javascript date object* representing the date the book was published

## Bonus

**App**

- Use localstorage (http://www.w3schools.com/html/html5_webstorage.asp) and JSON.stringify to save the state of your library

- Add a more robust search function to your app to allow you to filter by one or more book properties
    - the search function should return an array of book instances
- **Make your library a singleton**
    - **A prototyped book class should also be made, with each ‘book’ in your library being an instance of the book class.**