An object-oriented book-list!

- Create a class BookList

- Create another class called Book

- **BookLists** should have the following properties:

  – Number of books marked as read
  – Number of books marked not read yet
  – A reference to the next book to read (book object)
  – A reference to the current book being read (book object)
  – A reference to the last book read (book object)
  – An array of all the Books

- Each **Book** should have several properties:

  – Title
  – Genre
  – Author
  – Read (true or false)
  – Read date, can be blank, otherwise needs to be a JS Date() object

- Every **Booklist** should have a few methods:

  – .add(book)
    - should add a **book** to the books list.
  – .finishCurrentBook()
    - should mark the **book** that is currently being read as "read"
    - Give it a read date of new Date(Date.now())
    - Change the last **book** read to be the book that just got finished
    - Change the current **book** to be the next book to be read
    - Change the next **book** to be read property to be the first unread book you find in the list of books

- **Booklists** and **Books** might need more methods than that. Try to think of more that might be useful.

Interfaz para "The Reading List"

En este ejercicio vamos a comenzar a dotar de interfaz a nuestros proyectos. Para ello vamos a usar el código de la clase BookList del tema anterior (Unidad 3, Worksheet #5, The Reading List)

Crea una interfaz que permita probar la funcionalidad completa de las clases diseñadas. Tienes libertad para decidir el estilo pero trata de hacer éste lo más atractivo posible.

Ejemplo de Interfaz