

## Enunciado

Este ejercicio se hace de forma individual.

Se pretende realizar una implementación simplificada de un sistema gestor para un club de lectura.

El club consta de un conjunto de libros y de socios 'Lectores' que cogen libros, los leen y los retornan. En este ejercicio nos vamos a centrar sólo en los actores Libro y Lector. A continuación se indica que atributos y servicios tiene cada uno.

El alumno debe subir a Moodle los siguientes archivos: Libro.java y Lector.java.

Al alumno se le proporcionan dos programas de prueba uno para Libro (TestLibroAlum.java) y otro para lector (TestLectorAlum.java). Las pruebas no son exhaustivas. No se deben modificar las pruebas que hay en estos archivos, pero si se pueden añadir nuevas pruebas después de la última que haya en que da uno de los archivos.

## Clase Libro

Esta clase representa un libro. La información que representa un libro (estado) es:

- titulo: de tipo String que contendrá el título del libro.
- autor: de tipo String que contendrá el nombre y apellidos del autor (para este problema se asume que cada libro sólo tiene un autor)
- fechaPublicacion: Atributo de tipo fecha, que contendrá la fecha en que fue publicado
- nVecesPrestado: número entero que lleva la cuenta de las veces que ha sido prestado

Los servicios que ofrecen las instancias de la clase libro son:

- Constructor, inicializa los atributos necesarios. Tenga en cuenta que java establece valores por defecto a los atributos del objeto. El constructor no debe inicializar aquellos atributos que estén correctamente inicializado con el valor por defecto.
- Se ofrecerá gets de todos los atributos.
- Se ofrecerá un servicio llamado 'prestado' que incrementará en uno el número de veces que el libro ha sido prestado.
- Un método esIgual que indicará si el libro es igual a otro pasado como parámetro. Dos libros son iguales si y sólo si coincide el título, el autor y la fecha de publicación.
- Un método toString que retorna un String con el siguiente formato: "Título, Autor, leído: nnn" donde nnn es el número de veces que el libro ha sido leído

## Clase Lector

Esta clase representa al lector y estará formada por los siguientes atributos:

- nSocio: entero que representa el número de socio
- nombre: Contiene el nombre del socio
- historicoLectura: Vector (array) que mantiene un histórico de los últimos 10 libros que el lector ha leído. En la posición cero está el libro que hace más tiempo se leyó y en la última posición ocupada el libro que se ha leído más recientemente.

Los servicios que ofrecen las instancias de la clase Lector son:

- Constructor, inicializa los atributos necesarios. El constructor **no** recibirá información sobre el histórico de libros leídos ya que este histórico se irá rellenando durante la vida del objeto.
- Gets para número de socio y nombre.
- listarHistorico: Retorna una cadena de caracteres en la que se muestran por orden de antigüedad (primero el que se ha leído hace más tiempo) los datos de los libros que se han leído de la siguiente forma: "Titulo, Titulo, ....,Titulo"
- leerLibro: Servicio que recibe una instancia de libro, lo inserta en el histórico y le indica al libro que ha sido prestado. Este método funciona de la siguiente forma:
  - o 1º Si el número de libros leídos por el lector es menor que la capacidad del histórico este se inserta al final.
  - o 2º Si el número de libros leídos supera la capacidad del histórico se descarta el libro que hizo más tiempo que se leyó (se desplazan los libros una posición hacia delante dejando un hueco al final)
- esIgual: Se considera que dos lectores son iguales cuando coincide el número de socio y los libros leídos que hay en el histórico son los mismos y están en el mismo orden