

¿Qué pasa si desde el main llamamos al método descripción? ¿Por qué?

El programa dará error ya que el método descripción es una función privada de Libro.java por lo que el main, que se encuentra en BibliotecaApp.java, no podrá llamar a la función.

¿Podemos añadir varias veces el mismo objeto Libro a una lista? ¿Qué sucede si prestamos uno de ellos?

Sí, podemos añadir un objeto a una lista tantas veces como queramos. Pero no va a haber 2 libros, van a haber 2 referencias a un mismo objeto por lo que si prestamos uno de ellos se restara en ambos.

¿Qué pasa si eliminamos el método toString de la clase Libro? ¿Qué se imprime en el main?

Si lo eliminamos se perderá la posibilidad de consultar la información de un libro, en el main se imprime:

```
src.Libro@251a69d7
src.Libro@7344699f
src.Libro@6b95977
[src.Libro@251a69d7,           src.Libro@7344699f,           src.Libro@6b95977]
[src.Libro@251a69d7,           src.Libro@7344699f,
src.Libro@7e9e5f8a]
```

Su funcionamiento normal es:

```
"ISBN: 1. 'El Quijote' de Miguel de Cervantes [Disponible] (5 ejemplares disponibles)
ISBN: 2. 'El murciélagos' de Jo Nesbo [No Disponible] (0 ejemplares disponibles)
ISBN: 3. 'Learn Java' de David Hoffman [Disponible] (6 ejemplares disponibles)
[ISBN: 1. 'El Quijote' de Miguel de Cervantes [Disponible] (5 ejemplares disponibles),
ISBN: 2. 'El murciélagos' de Jo Nesbo [Disponible] (1 ejemplares disponibles), ISBN: 3.
'Learn Java' de David Hoffman [Disponible] (6 ejemplares disponibles)]
[ISBN: 1. 'El Quijote' de Miguel de Cervantes [Disponible] (5 ejemplares disponibles),
ISBN: 2. 'El murciélagos' de Jo Nesbo [Disponible] (1 ejemplares disponibles), ISBN: 3.
'Learn Java' de David Hoffman [Disponible] (6 ejemplares disponibles), ISBN: 4. 'Con
viento solano' de Ignacio Aldecoa [Disponible] (1 ejemplares disponibles)]"
```

¿Podemos crear un Libro omitiendo el número de ejemplares disponibles en la llamada a su constructor? ¿Cómo podemos modificar el programa para lograrlo?

No, porque el programa lo requiere en todos sus constructores para Libro, sin embargo, podríamos añadir otro constructor que contemple la posibilidad de que se omita dicha información y le asigne un valor predefinido.

¿Cómo has realizado tu diseño para que el filtrado por categorías sea eficiente? Justifica tu respuesta

La clase biblioteca guarda su nombre y un HashMap donde la clave es el nombre de un género y el valor es una lista con todos los libros de ese género, haciendo que, en vez de recorrer una lista con todos los libros, buscando aquellos que tengan el mismo género, solo tenga que devolver la lista que le corresponde al género que buscamos.

¿Cómo podríamos modificar el diseño para tener géneros jerárquicos (es decir, géneros dentro de géneros como, por ejemplo, Novelas con hijas Novela.Historica y Novela.CienciaFiccion)?

Se podría meter un HashMap dentro de otro HashMap, en el primero se tendría los géneros padres como podría ser Novela y que este llevaría a otro HashMap que tuviese como claves los géneros hijos como Novela.Historica y Novela.CienciaFiccion siendo su valor una lista con todos los libros que tengan ese género.

¿Como podríamos hacer el diseño eficiente para obtener todos los libros escritos por un autor? En este diseño, queremos añadir información a cada autor, como su año de nacimiento y nacionalidad. Justifica tu respuesta.

Se podría crear una nueva clase Autor que tenga todos los datos del autor pertinentes y añadimos en cada libro un campo llamado autor de clase Autor, para que hacer una búsqueda eficiente se haría otro HashMap que tenga como clave el nombre del autor y como valor una lista con todos sus libros creados, consiguiendo lo mismo que con los géneros, que en vez de caminar por todos los libros solo tengamos que sacar su lista.