

# Cuestiones:

Alejandro González y Fernando Blanco

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

Si desde el main llamamos al método description, a la hora de compilar el código, nos dará error, esto sucede porque description es un método privado al que sólo se puede acceder desde la clase objeto.

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

Sí podemos añadir varias veces el mismo objeto Libro a una lista, si prestamos uno de ellos, se reducirá la cantidad de ejemplares disponibles del objetos que hemos prestado, pero no del otro objeto.

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

Si eliminamos el método toString de la clase Libro, ya no tendremos un método que sobrescriba al método toString de Object, y se imprimirá en el main la información del objeto, siguiendo el método de Object, la información aparecerá ilegible.

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

No se puede crear, para evitarlo habría que quitar del constructor uno de los campos (ejemplaresDisponibles).

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

Agrupamos los libros en un Map<String, List<Libro>> donde la clave es el género. Así, librosPorGenero() hace una búsqueda directa en tiempo O(1) promedio HashMap sin recorrer toda la colección.

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

Podríamos representar el género como una ruta, por ejemplo, "Novela/Historia" y mantener también un árbol de géneros (`Map<String, List<String>>` de padres a hijos) o una estructura tipo nodo. Al filtrar por un género padre, se incluirían los libros de ese género y de todos sus descendientes.

¿Cómo podríamos hacer un 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.

Añadiría una clase Autor con campos nombre, año nacimiento, nacionalidad y usaría un `Map<Autor, List<Libro>>` o `Map<String, Autor> + Map<Autor, List<Libro>>` para acceso rápido. Así, obtener libros por autor sería  $O(1)$  promedio y además se mantiene la información adicional del autor en un solo lugar.