

# **TAREA 7:**

## **Ocultación de la información**

**Lernik Gasparyan**



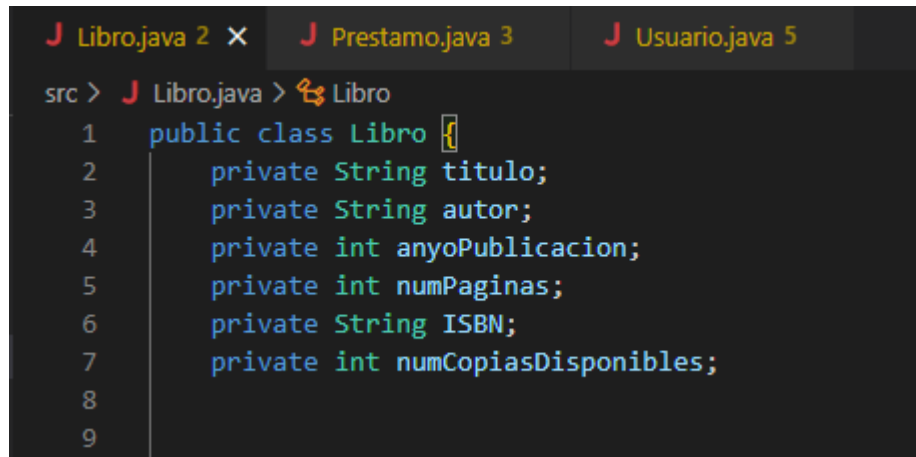
**ENTORNOS DE DESARROLLO**  
GRADO SUPERIOR EN DESARROLLO DE APLICACIONES WEB  
2023-2024

## Índice

Informe de justificaciones.....	3
Mecanismos de visibilidad.....	4
Bibliografía.....	5

## Informe de justificaciones

He restringido el acceso a todos los atributos, definiéndolos como privados para ocultar su información y uso interno dentro del código.



```
src > J Libro.java > Libro
1  public class Libro {
2      private String titulo;
3      private String autor;
4      private int anyoPublicacion;
5      private int numPaginas;
6      private String ISBN;
7      private int numCopiasDisponibles;
8
9
```

Se accederá a los atributos a través de los métodos, que serán públicos, de esta manera garantizamos el control sobre la manipulación de datos.

Al ser públicos los métodos, se puede acceder a estos desde cualquier otra clase.



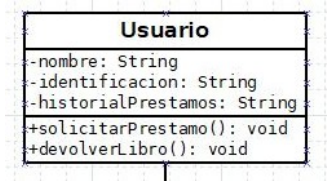
```
public void gestionarPrestamo(){
    if (numCopiasDisponibles > 0) {
        numCopiasDisponibles--;
        System.out.println("Prestado. Copias disponibles: " + numCopiasDisponibles);
    } else {
        System.out.println(x:"No hay copias del libro.");
    }
}

public void devolucionLibro(){
    numCopiasDisponibles++;
    System.out.println("Libro " + titulo + " devuelto.");
}
```

De esta manera abstraemos al resto del mundo de la complejidad de la implementación interna, además así se expone el estado del objeto sólo a través de los métodos que hayamos declarado públicos.

Logramos evitar cambios no deseados, obtenemos facilidad de cambios internos y menor grado de dependencia.

Por supuesto, todo dependerá del motivo de uso de la aplicación.



## Mecanismos de visibilidad

Pudiendo en este ser conveniente tener ciertos métodos o atributos públicos, privados o protegidos, pero de forma general se indican públicos los métodos.

Publico (Public) se podría acceder desde cualquier clase o cualquier parte del programa.

Privado (Private) solo se podría acceder desde operaciones de la misma clase.

Protegido (Protected) solo se podría acceder desde la misma clase o clases derivadas, por la herencia.

Tipo de Acceso Se accede desde	PRIVATE	PROTECTED	PUBLIC
La misma clase	Sí	Sí	Sí
Clases hijas	No	Sí	Sí
Clases externas	No	No	Sí

Con la ocultación de la información por lo tanto logramos ocultar detalles de implementación y prestar al exterior el interfaz mínimo, es decir, métodos y propiedad que se pueden llegar a ver.

## Bibliografía

Lucid Software Español. (2019, 4 febrero). *Tutorial - Diagrama de clases UML* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=Z0yLerU0g-Q>

Vucic, M. (2024, 30 enero). *Encapsulación en Java*. CodeGym. <https://codegym.cc/es/groups/posts/es.98.encapsulacion-en-java>