

PRÁCTICA 2 DE JAVA PROGRAMADOR I

Tienes que escribir un programa de consola que permita editar un libro formado por un título y por varios capítulos.

Cuando arranque, el programa debe solicitar el título de libro y cuántos capítulos tendrá como máximo.

A continuación debe mostrar el siguiente menú:

```
MENÚ PARA EL CAPÍTULO __ DE __
-----
1-Elegir un nuevo capítulo.
2-Ver el contenido del capítulo actual.
3-Añadir texto al capítulo actual.
4-Buscar capítulos que contienen una palabra.
5-Salir
Introduce una opción (1-5): _
```

Al final de la primera línea del menú debe mostrarse un número de capítulo y el número máximo de capítulos. Por ejemplo:

```
MENÚ PARA EL CAPÍTULO 1 DE 10
```

La primera vez se seleccionará el capítulo 1 por defecto.

Para cada opción se deben ejecutar las siguientes tareas:

- Opción 1: Solicitará un número de capítulo válido.
- Opción 2: Se mostrará el contenido del capítulo si tiene algo escrito o el texto "<No hay nada escrito>" si todavía no se escribió nada.
- Opción 3: El usuario podrá escribir un texto y cuando salte de línea se añadirá como párrafo al final del capítulo actual.
- Opción 4: Debe solicitar una palabra y mostrará el número de aquellos capítulos que incluyan la palabra literalmente.
- Opción 5: El programa finalizará.

Después de ejecutarse cada tarea debes volver a mostrarse el menú, excepto para la opción 5.

Procedimiento:

Crea una clase llamada **Menu** que contenga métodos para realizar cada operación de la interfaz de usuario: mostrar el menú y mantenerlo mediante un bucle, leer un dato numérico y una línea de texto, y cualquier otra operación requerida.

Crea una clase **Libro** aplicando los patrones de diseño de clases habituales y el concepto de encapsulación. El diseño de esta clase debe realizarse sin pensar en la interfaz de usuario que se aplicará a través de la clase **Menu**.

La clase Libro debe contener, al menos, un campo string para guardar el título del libro, un array de strings para almacenar los capítulos, y un número entero que indique el capítulo seleccionado.

Crea las propiedades y métodos adecuados, de forma que se puedan realizar las operaciones de manipulación de capítulos de forma segura. Si a través de un parámetro se hace referencia a un capítulo no válido se debe trabajar con el primer capítulo.