

## Ejercicio 3. Arreglos dinámicos

### Análisis

#### Requisitos funcionales:

- Debe de permitir registrar los datos de los libros.
- Debe de registrar los préstamos de libros.
- Debe de permitir registrar manualmente los datos de los libros y miembros.
- Debe de permitir agregar nuevas sucursales de la biblioteca.
- Debe de almacenar la información separada para cada sucursal.
- Debe de mostrar la cantidad total de libros prestados en un determinado periodo de tiempo.
- Debe de mostrar los géneros de libro más solicitados.
- Debe de mostrar el libro más prestado.

#### Propósito de cada clase:

Géneros: Gestiona una colección de géneros literarios.

Libro: Su propósito es almacenar los datos del libro.

Miembro: Su propósito es almacenar los datos del miembro y libros prestados.

Sucursal: Su propósito es almacenar la información de la sucursal, de los libros de esta sucursal y de los miembros de esta.

Prestamo: Su propósito es realizar todos los métodos para realizar los préstamos de libros.

DriverProgram: Se utilizará para mostrar lo necesario al administrador.

#### Propósito de atributos:

Géneros:

- Géneros: `ArrayList<String>`: Lista de géneros literarios.

Libro:

- ISBN: `String`. Su propósito es almacenar el ISBN del libro.
- Titulo: `String`. Su propósito es almacenar el título del libro.
- Autor: `String`. Su propósito es almacenar el nombre del autor del libro.
- Año: `int`. Su propósito es almacenar el año de publicación del libro.
- Género: `String`. Su propósito es almacenar el género del libro.
- Fechap: `Date`. Su propósito es almacenar la fecha en la que se presta el libro.
- Fechad: `Date`. Su propósito es almacenar la fecha de devolución del libro.
- Vecesp: `int`. Su propósito es almacenar la cantidad de veces que se ha prestado el libro.

Miembro:

- ID: `int`. Su propósito es almacenar el ID del miembro.

- Nombre: String. Su propósito es almacenar el nombre del miembro.
- Librospm: ArrayList<Libro>: Su propósito es almacenar los libros prestados por el miembro.

Sucursal:

- Nombre: String. Su propósito es almacenar el nombre de la sucursal.
- Libros: ArrayList<Libro>: Su propósito es almacenar los libros de esa sucursal.
- Miembros: ArrayList<Miembro>: Su propósito es almacenar los miembros de esa sucursal.
- Librosd: ArrayList<Libro>. Su propósito es almacenar los libros disponibles.
- Librosp: ArrayList<Libro>. Su propósito es almacenar los libros prestados.

Prestamo:

- Sucursales: ArrayList<Sucursal>. Su propósito es almacenar las sucursales de las bibliotecas.

DriverProgram:

- Scanner: Scanner. Utilizado para la entrada de datos del usuario.

### **Propósito de métodos:**

Géneros:

- agregarGenero(): Añade un nuevo género a la lista.

Libro:

Miembro:

- agregarLibrosPM(Libro): void. Su propósito es agregar los libros que prestó a la lista de sus libros prestados.

Sucursal:

- agregarLibrosP(Libro): void. Su propósito es agregar los libros que prestó a la lista de libros prestados.
- agregarLibrosD(Libro): void. Su propósito es agregar los libros que están disponibles a la lista de libros disponibles.
- agregarLibros(Libro): void. Su propósito es agregar los libros de la sucursal a la lista de libros.
- agregarMiembros(Miembro): void. Su propósito es agregar los miembros de la sucursal a la lista de miembros.
- prestarLibro(String isbn, Miembro miembro, Date fechaPrestado, Date fechaDevolucion): Gestiona el préstamo de un libro a un miembro.
- cantidadLibrosP(): Retorna la cantidad de libros prestados.
- generosMasSolicitados(): Determina el género más solicitado.
- libroMasPrestado(): Identifica el libro más prestado.

Prestamo:

- agregarSucursales(Sucursal): void. Su propósito es agregar las sucursales.

DriverProgram:

- Main(args String[]): Se utilizará para mostrar al usuario lo necesario.
- prestarLibro(ArrayList<Sucursal> sucursales): Maneja el proceso de préstamo de un libro.
- verEstadisticas(ArrayList<Sucursal> sucursales): Muestra las estadísticas de una sucursal seleccionada.

## **Diseño**

