

PROGRAMACIÓN

Prueba de Nivel 1 - Examen 3ª Evaluación

Indicaciones para el examen.

- Cada uno de los apartados, se deberá entregar en una clase o fichero distinto. El nombre de esta clase empezará con tus iniciales (Apellidos+Nombre) y el ejercicio con su número (pero sin el punto). Ejemplo: El ejercicio 1.1, en un fichero con nombre MGCJEjercicio11.java => Se penalizará con 0,1 puntos cada fichero que no cumpla esta regla
- Se deben respetar las reglas de estilo de Java para clases, objetos, variables,... => Se penalizará con 0,1 puntos cada vez que no cumpla esta regla.
- Para ser calificado de forma correcta, cada apartado deberá ir acompañado de un pequeño código ejecutable, que demuestre su funcionamiento.
- (0,5 puntos) El código deberá ir correctamente tabulado y comentado (solo aquellas partes del código de especial dificultad)

1. Biblioteca.

Queremos hacer una aplicación que lleve un listado de los ejemplares de una biblioteca. Se dividen en tres grupos: Libros, Películas y Música

Para gestionar todos los ejemplares, se programarán las siguientes clases con su herencia:

Ejemplar
String codigo String titulo String autor LocalDate fechaPublicacion LocalDate fechaPrestamo LocalDate fechaDevolucion static int numeroEjemplares
- Ejemplar (String titulo, String autor, LocalDate fechaPublicacion) => Constructor. El código se genera automáticamente a partir del número de ejemplares y siempre debe tener 6 posiciones, completando con ceros por la izquierda (Ej: "Bib000001", "Bib000002", "Bib000003", ...) - alquilar() => Asigna valores a la fechaAlquiler y fechaDevolucion - mostrar() => Muestra titulo y autor del ejemplar, incluyendo los años desde su publicación. - mostrarAmpliado() => Muestra todos los datos del ejemplar, incluyendo las fechas de alquiler y devolución.

NOTA: Elige la forma más adecuada, para guardar el nombre de la Biblioteca "BibVentura".

IMPORTANTE: Es posible que sea necesario añadir otros métodos como algún get/set, equals,....

Sus clases hijas:

Libro

Pelicula

Musica

- Libro => Tiene un atributo propio que es el número de páginas. Se muestra con el formato reducido.
- Pelicula => Tiene un atributo propio que es la duración de la película. Se muestra con el formato reducido.
- Musica => Tiene un atributo propio que es el número de canciones del disco. Se muestra con el formato reducido.

Gestión de los alquileres de ejemplares Ampliar el programa anterior. Añadiendo:

- Los atributos **fechaAlquiler** y **fechaDevolucion** (ambos de tipo `LocalDate`), inicialmente serán **null**.
- El método en la clase Ejemplar, de nombre **alquilar()**. Este método pone la fecha del día actual en **fechaAlquiler** y genera la fecha de devolución. Para ello, este método será sobrescrito en las clases hijas, para escribir la **fechaDevolucion**, de la siguiente forma:
 - Para la clase Libro, el valor de **fechaDevolucion** será un mes después de la **fechaAlquiler**.
 - Para la clase Pelicula, el valor de **fechaDevolucion** será dos semanas después de la **fechaAlquiler**.
 - Para la clase Musica, el valor de **fechaDevolucion** será cinco días después de la **fechaAlquiler**.
- El método **mostrar**, en la clase Ejemplar, también debe ser reescrito en las clases hijas, para incluir a mano el tipo de ejemplar.
Libro : Titulo "La Comunidad del Anillo" Autor:"Tolkien"
- El método **mostrarAmpliado**, en la clase Ejemplar, que muestre todos los datos de los ejemplares (puede llamar a su vez a **mostrar()**). Debe tener en cuenta:
 - Si el ejemplar, no está alquilado ahora mismo, se mostrará un mensaje: **"El ejemplar está disponible para alquilar"**.
 - Si el ejemplar, está alquilado ahora mismo, se incluirá las fechas de alquiler y devolución en formato "17 de Mayo de 2023"

Programa principal

Para hacer uso de estas nuevas funcionalidades, el programa principal deberá ampliarse y manejar dos listas distintas, que deberán ser creadas en la función main (no se admitirán como globales):

- listaEjemplares => Lista con todos los ejemplares que no están alquilados en este momento. Se puede inicializar con varios ejemplos al final del ejercicio.
- listaEjemplaresAlquilados => Lista con todos los ejemplares que están alquilados en este momento. Inicialmente estará vacía.

El programa tendrá el siguiente menu (controlar los posibles errores si se introduce una opción incorrecta):

- 1- Mostrar listas. Muestra la listaEjemplares y listaEjemplaresAlquilados, con información básica.
- 2- Mostrar listas ampliadas. Muestra la listaEjemplares y listaEjemplaresAlquilados, con información ampliada, sobre sus fechas de alquiler y/o devolución; y mostrando un ejemplar en cada línea.
- 3- Alquilar ejemplar. Gestiona el alquiler de un ejemplar. Solicita un código de ejemplar por pantalla. Controlar con una expresión regular que es un código correcto, o volver a solicitarlo.
 - Si el código no está en listaEjemplares => Mostrar un aviso **"El ejemplar indicado, no está en la lista de disponibles"**
 - Si el código está en listaEjemplares => Se actualizan sus datos de fechaAlquiler y fechaDevolución, y se mueve a listaEjemplaresAlquilados (borrándolo de listaEjemplares).
- 4- Crear nuevos ejemplares. Pide por pantalla los datos de un ejemplar nuevo y lo añade en listaEjemplares. Para ello deberá preguntar si es un Libro, Pelicula o Musica. Bastará con pedir por pantalla titulo, autor, fechaPublicación y el dato propio de cada uno. Es necesario controlar que la fecha de publicación sea correcta, y volver a solicitarla, en caso contrario.
- 5-Salir

Ejemplos para crear ejemplares de distintos tipos:

```
Libro libro1 = new Libro("La Comunidad del Anillo", "Tolkien", LocalDate.of(1954, 07, 25), 5000);
Musica musical = new Musica("Sultans of Swing", "Dire Straits", LocalDate.of(1978, 07, 02), 18);
Pelicula pelicula1 = new Pelicula("Avengers: Endgame", "Marvel", LocalDate.of(2019, 04, 26), 220);
```