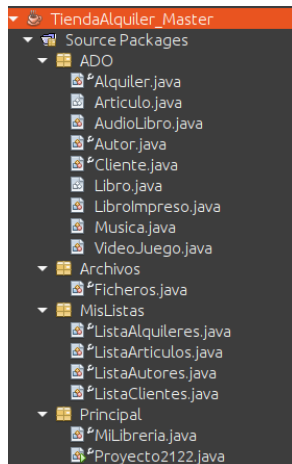


## Q10 - GESTIÓN DE TIENDA DE ALQUILER

Vamos a avanzar en el ejercicio desarrollado en una quincena anterior, puedes descargarlo en un carpeta de esta quincena, este es el esquema **obligatorio** de clases:



**IMPORTANTE:** Podéis utilizar clases propias, pero estas deben estar en el proyecto y deben ser las especificadas en la carpeta de esta quincena.

### Clase Articulo:

- Clase abstracta, de ella heredan las clases Libro, Musica y VideoJuego.
- Tendrá los siguientes atributos de clase y atributos miembro (los que yo pongo son obligatorios) si alguien quiere poner más puede comentarlo o ponerlos directamente.

// atributos de clase

```
private static float preciobase = 1;
```

// atributos miembro

```
private String codArt;  
private String nombre;  
private String fecha;  
private int anyo;  
private String resumen;  
private String autor;  
private boolean deteriorado;  
private ArrayList<String> comentarios = new ArrayList();
```

- También definirá **tres métodos abstractos**, uno para mostrar datos, otro para calcular el alquiler de un artículo para un determinado número de días y otro para guardar comentarios.

```
public abstract String info();  
public abstract float precioAlquilerArticulo(int dias);  
public abstract boolean guardaComentario(String c);
```

- Además de sus constructores, gets sets ...

### Clases Musica y VideoJuego

```
public class Musica extends Articulo{  
    public static float suplemento = 2f;  
  
    private int nPistas;  
    private float duracion;  
    private String tipo; // cd o vinilo  
    .....
```

```
public class VideoJuego extends Articulo {  
  
    public static float suplemento = 5f;  
  
    private int edadRecomendada;  
    private String plataforma; // PS2, PS3, PS4, XBOX, PC,...  
    private String categoria; //Acción, Deportivo, Estrategia, Simulación,...  
    private boolean formatoFisico;  
    .....
```

## Clase Libro

- Clase abstracta, de ella heredan las clases **LibroImpreso** y **AudioLibro**
- Tendrá un solo atributo:

```
public abstract class Libro extends Artículo {  
    // atributos miembro  
    private int numPaginas;
```

## Clases LibroImpreso y AudioLibro

### Clase **LibroImpreso**:

```
public class LibroImpreso extends Libro{  
    private boolean formato;  
    private String tapa;  
    .....
```

### public class **AudioLibro** extends **Libro** {

```
    private boolean sintetizador;  
    private int horasDuracion;  
    private int minDuracion;  
    .....
```

## Clase **Autor**

```
public class Autor {  
    // atributos de Autor  
    private String nom;  
    private String Apell;  
    private int edad;  
    private int publicaciones;
```

## Clase Cliente (socios de la tienda)

```
public class Cliente {  
    /* Atributos de clase: un int que permita luego asignar un número  
       correlativo al código de socio, impidiendo que se repita.  
    */  
    private static int num;  
  
    // Atributos miembro  
    private final String codSocio; // final porque no va a variar una vez que se asigna a un socio ya no varía  
    private String dni;  
    private String nombre;  
    private String apellidos;  
    private String direccion;  
    private String telf1;  
    private String telf2;  
    private String telf3;  
    private int puntos;
```

## Clase Alquiler (Tiene tres atributos: Un Artículo, un Socio (cliente) y los días de alquiler)

```
public class Alquiler {  
    // Atributos  
    private Artículo a;  
    private int dias;  
    private Cliente cli;
```

## Clase **ListaArticulos**

Para la clase **ListaArticulos** vamos a tener un solo atributo, el ArrayList de **Articulos**

### **SIN GET NI SET DEL ArrayList, + CONSTRUCTORES, Y MÉTODOS**

```
public class ListaArticulos {  
    private ArrayList<Articulo> listaArt;  
    public ListaArticulos() {  
        this.listaArt = new ArrayList();  
    }  
    public ListaArticulos(ArrayList<Articulo> listaArt) {  
        this.listaArt = listaArt;  
    }  
}
```

.....

Los siguientes métodos:

- Constructores.
- public boolean **anyadirArticulo**(Articulo art) ...
- public boolean **borrarArticulo**(Articulo art) ...
- public boolean **borrarArticulo**(String codArt) ...
- public boolean **modifArticulo**(String codArt, ..... ) ...
- public Vehiculo **buscaArticulo**(String codArt) ...
- public boolean **existeArticulo**(String codArt) ...
- public boolean **vacialista**() ...
- public int **numArticulos**() ...
- public void **mostrarPrecios**(int dias) ...
- public void **mostrarArticulo**() ...
- public void **muestraTipo**(String tipo) throws ClassNotFoundException ...
- .....

## Clases **ListaClientes**, **ListaAutores** y **ListaAlquileres**

Son muy parecidas a la clase ListaArticulos ....

Crea una carpeta llamada Archivos para almacenar los ficheros.

## Clase **Ficheros**

Crea una clase llamada **Ficheros** con dos métodos estáticos para el manejo de los ficheros:

- public static boolean **guardarArticulos**(ListaArticulos listaArt)
- public static boolean **cargarArticulos**( ListaArticulos listaArt)

## Clase principal main()

**Se pide:** mejorar el programa principal en las opciones que correspondan para:

- Alta de Artículos. **(los mensajes de error serán lo más precisos posibles)**
  - Se podrá dar de alta un LibroImpreso, un AudioLibro, una Musica, y un VideoJuego.
  - Control de codArt: se creará de forma automática con tres letras y 4 números. Las tres letras serán LIM, LAU, MUS y VID, seguidos de la numeración. Por ejemplo par aun audiolibro podría ser LAU0025
  - Controlar el tipo de música (CD-Vinilo)
  - Controlar el plataforma y categoría permitida en VideoJuego
  - Controlar que el año sea  $\geq 2000$  y  $\leq 2050$
  - .....
- Escribir opciones para el mantenimiento de la colección de Articulos, Bajas, Consulta, Modificación.
- Escribir en un fichero todos los Artículos de la colección de artículos.
- Recorrer el fichero para cargar los artículos guardados en el objeto ListaArticulos

Opcional:

- Guardar en un fichero los Alquileres y recuperar los datos.
- Guardar en un fichero los Clientes y recuperar los datos.
- Guardar en un fichero los Autores y recuperar los datos.

El menú queda con las siguientes opciones:

```
System.out.println("¿Qué quieres hacer?\n");
System.out.println("1. Mantenimiento artículo (Alta, Baja, Consulta, Modificación ");
System.out.println("2. Precio alquiler artículo");
System.out.println("3. Precio alquiler todos los artículos");
System.out.println("4. Alta de un cliente (SOCIO)");
System.out.println("5. Alta de un autor");
System.out.println("6. Alquilar un artículo");
System.out.println("7. Devolver un artículo");
System.out.println("8. Cargar Ficheros");
System.out.println("9. Guardar Ficheros");
System.out.println("0. Salir\n");
```