

Práctica 17: Eventos DVDClub

En esta práctica le vamos a desarrollar un interfaz gráfico al ejercicio de la práctica 9 que controlaba el funcionamiento de un club de DVD's, vamos a seguir para la organización del problema el Modelo Vista Controlador (MVC).

Pasos a seguir:

- 1.- Necesitamos las clases que se hicieron en la práctica 9: **DVD**, **Ciente**, **CienteDinero** y **DVDClub**. Tenemos dos opciones, podemos importar las que hicimos y que estarán en un paquete con nombre práctica8 o similar, o bien, para más comodidad, las copiaremos en este nuevo proyecto, os aconsejo que hagáis esto último, ya que haremos algunos cambios en la clase DVDClub.
- 2.- Crearemos nuestra propia clase de manejo de excepciones, así las situaciones excepcionales propias de la gestión del DVDClub se tratarán con esta nueva llamada **DVDClubException**. El código de esta clase podría ser:

```
public class DVDClubException extends RuntimeException {  
    public DVDClubException() {super();}  
    public DVDClubException(String msg) {super(msg);}  
}
```

- 3.- A la clase DVDClub, se le realizarán los siguientes cambios:

- Los datos relativos a los clientes se almacenarán en lugar de en una lista en un conjunto.
- Vamos a **sobrecargar el constructor**, el que ya tenemos, crear un conjunto de clientes y otro de películas en blanco. El nuevo constructor recibirá como argumentos los nombres del fichero de clientes y del fichero de películas con los que se ha de inicializar el DVDClub, es decir, en lugar de crear un DVDClub sin datos previos, lo haremos a partir de datos que tendremos almacenados en ficheros de texto.
 - Necesitamos importar el paquete `java.io`.
 - Los nombres de los clientes en el fichero de texto de clientes vendrán dados separados por espacios en blanco.
 - En el fichero películas, cada línea contendrá la información concerniente a una película, estando los distintos datos separados por asterisco ("*"); los actores vendrá dados separados por comas ordenados según las importancia de su papel en la película.
 - Tómense como ejemplo los ficheros `clientes.txt` y `peliculas.txt` proporcionados.
- El método "**cliente**" no debe distinguir entre mayúsculas y minúsculas a la hora de buscar el cliente. Para ello puedes usar el método `equalsIgnoreCase` en lugar de `equals`.
- En el método "**alquila**", un cliente no válido, un título no disponible o cualquier otra situación anómala deben producir excepciones, en lugar de devolver un valor booleano a falso, que era lo que hacíamos hasta este momento. A modo de ejemplo, nuestro código antes podía ser:

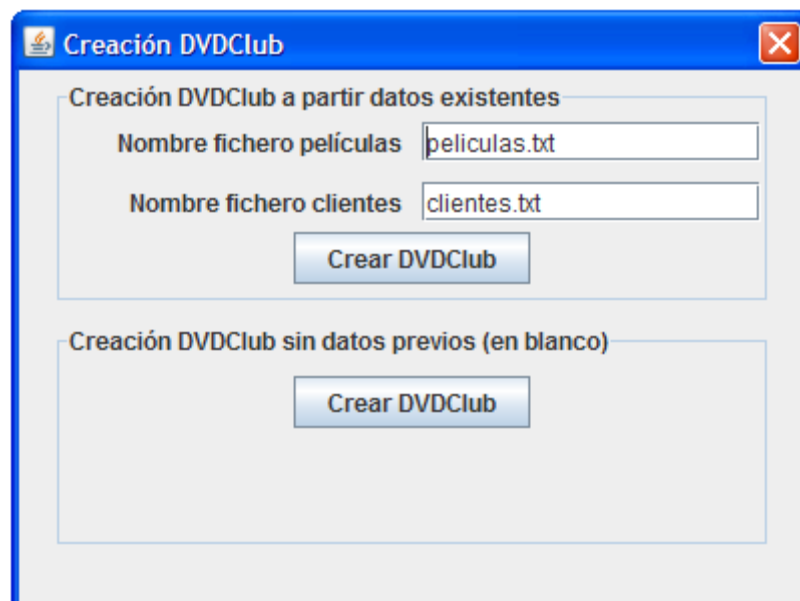
```
if (peli==null) return false;
```

Ahora lanzaremos una excepción, como la siguiente:

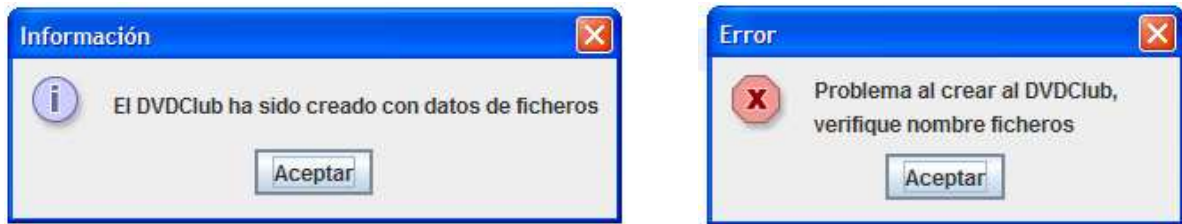
```
if (peli==null)  
    throw new DVDClubException("Película no disponible");
```

- En el método "**devuelve**", al igual que en el caso anterior deberán tratarse adecuadamente las situaciones excepcionales.

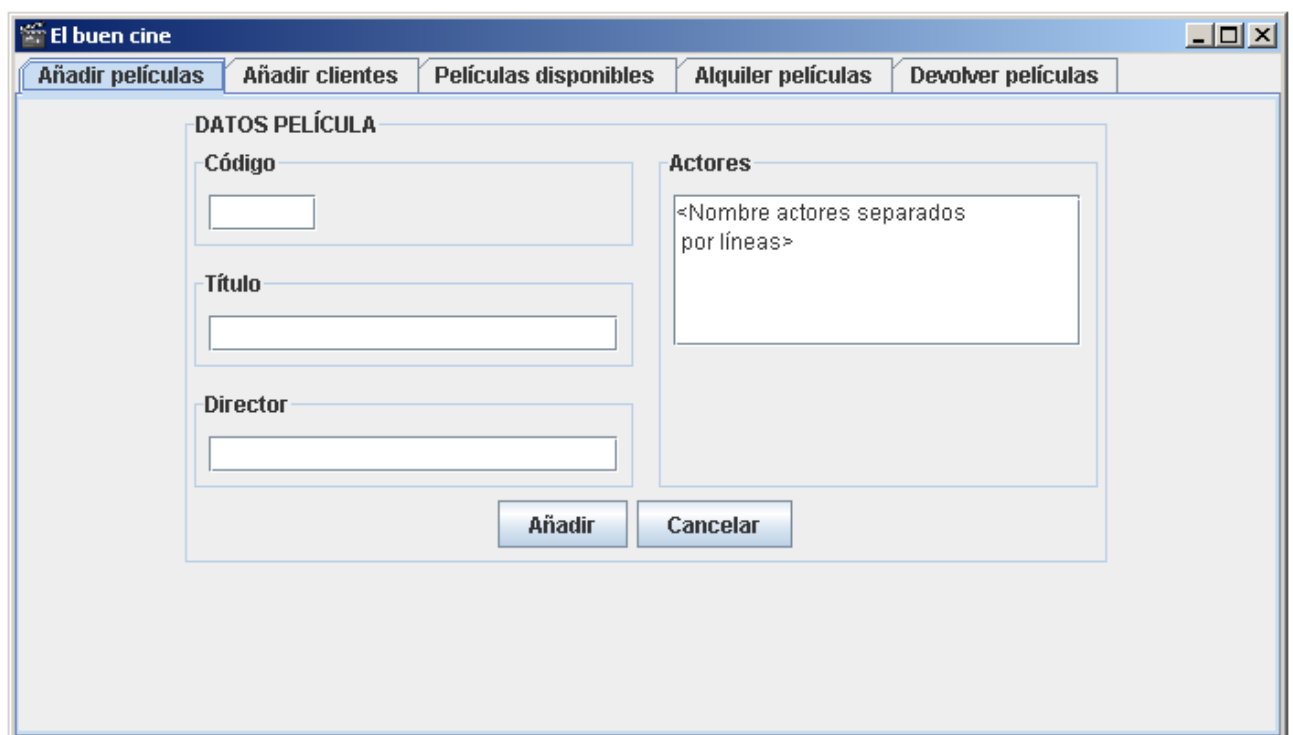
- Comprueba que tanto el método “*disponibles*” como el método “*alquiladas*” devuelven un String, en lugar de mostrar los datos en consola. Si no es así modifícalos. Estos métodos deben proporcionar sus salidas en un formato adecuado (“\n” para retorno de carro, “\t” para tabulador, ...). Por ejemplo, cada película debe estar separada de la anterior por un salto de línea “\n”.
 - Añade un método “*public SortedSet<DVD> copias(String titulo)*”, que dado un título devuelva el conjunto de DVD’s de la película disponibles (conjunto de DVDs con la misma película y distinto código). No se distinguirán mayúsculas y minúsculas.
 - Comprueba si tienes dos métodos para que devuelvan la lista de películas y la lista de clientes respectivamente, sino es así hazlos.
- 4.- Comprueba que tu ejercicio sigue funcionando correctamente tras los cambios, utiliza para ello el programa de prueba que hiciste en la práctica 9.
- 5.- [Opcional] Con respecto al **DVDClubRentable** vamos a desarrollar el ejercicio completo, para ello:
- Vamos a crear la clase DVDClubRentable, esta clase funcionará como un DVDClub pero teniendo en cuenta que el alquiler de cada película cuesta 2 €. El precio del alquiler como puede variar a lo largo del tiempo se almacenará en una variable que sea constante y de clase, ya que va a ser un precio único para todo el DVDClub.
 - En la clase **ClienteDinero** ya que estamos manejando excepciones, si un cliente no tiene saldo suficiente para alquilar una película, lanzaremos una excepción.
- 6.- Como hemos indicado anteriormente nuestra práctica sigue el MVC, el **modelo** podemos considerar que es lo que hemos desarrollado hasta ahora, por tanto nos falta la **vista** (interfaz gráfico), y el **controlador** (encargado de programar los eventos).
- 7.- A continuación vamos a empezar a diseñar un interfaz para nuestra aplicación, dicho interfaz podría ser como en el que se muestra a continuación, aunque también podéis cambiarlo a vuestro gusto.
- En primer lugar se lanzará un diálogo similar al siguiente, que nos permitirá poder crear el DVDClub a partir de datos almacenados en ficheros, o bien sin ningún dato previo. En caso de que el usuario pulse el botón de cierre nos saldremos de la aplicación (para ello deberemos capturar en el *main*, la excepción de puntero nulo, y hacer un `System.exit(0)`)



Tras pulsar el botón crear se mostrará otra ventana informando de si todo ha ido bien o no, en el caso de que todo vaya bien se mostrará un mensaje como el de la imagen de la izquierda, y en caso de que haya algún problema con los ficheros, uno como el de la imagen de la derecha.



Cuando desaparezca esta ventana, aparecerá la pantalla principal de la aplicación cuyo aspecto podría ser el siguiente:



Como también se aprecia la ventana principal de la aplicación, a parte de esa barra de menús, está organizada en pestañas, la forma de crear una pestaña

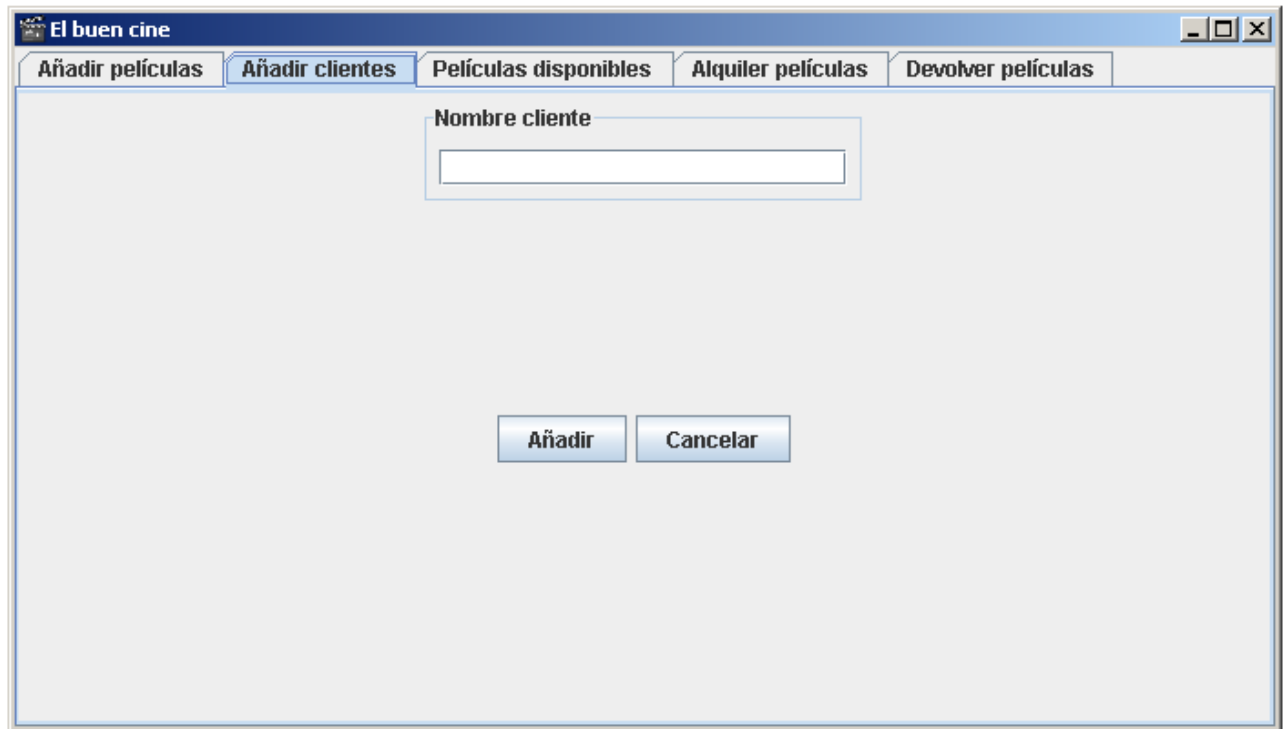
```
// Crear un TabbedPane (pestañas)
JTabbedPane tabbedPane = new JTabbedPane();

setLayout(new GridLayout(1, 1));
// Hay que darle este gestor de esquemas al panel sino pondrá una
// pestaña debajo de otra

// Añadir paneles
Component panel1 = panelPelículas();
tabbedPane.addTab("Añadir películas", panel1);
tabbedPane.setToolTipTextAt(0, "Permite añadir películas al DVDClub");
// Ponemos texto a la pestaña, para que se visualice al acercar el ratón
```

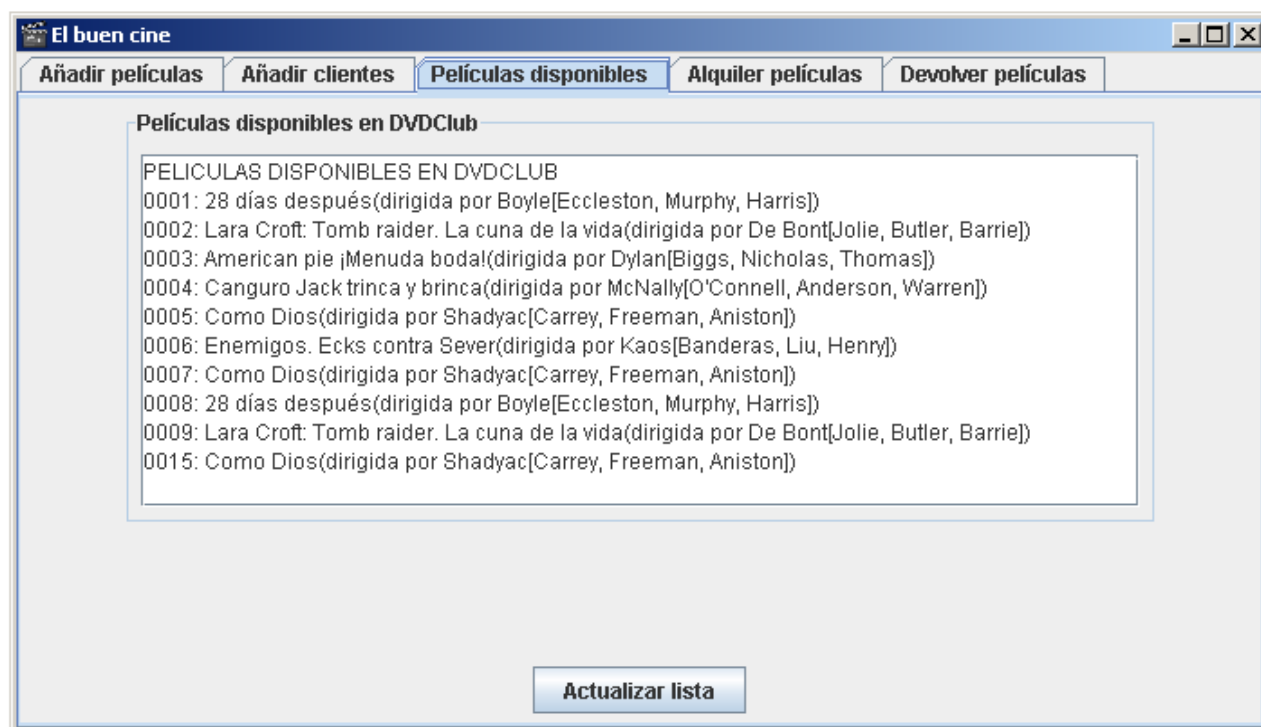
La primera pestaña ya se puede ver su aspecto en la imagen anterior, y lo que hace es recoger datos de una nueva película para añadirla al DVDClub, si se pulsa el botón Añadir, y si se pulsa el botón Cancelar, se inicializará el formulario.

La segunda pestaña, nos permitirá añadir clientes al DVDClub si se pulsa el botón Añadir, y si se pulsa el botón Cancelar, se inicializará el formulario. Su aspecto podría ser el siguiente:

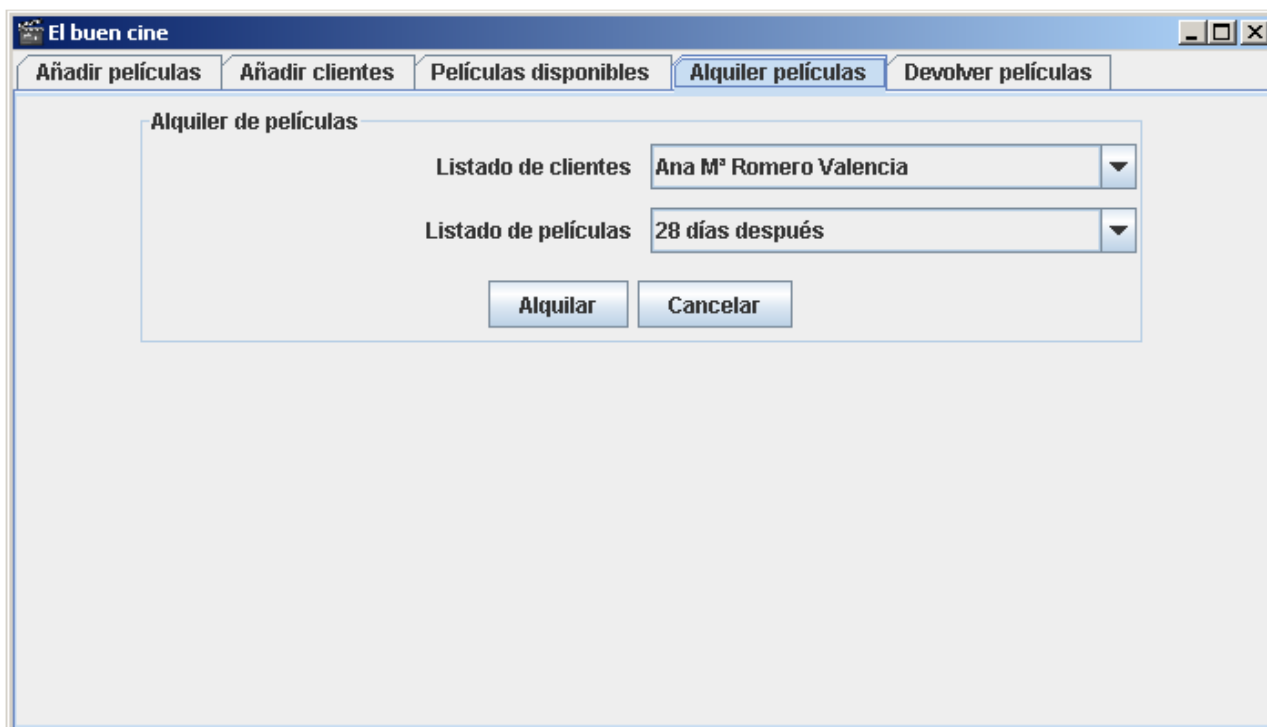


The screenshot shows a Java Swing window titled "El buen cine". It has five tabs: "Añadir películas", "Añadir clientes" (which is selected), "Películas disponibles", "Alquiler películas", and "Devolver películas". The "Añadir clientes" tab contains a text input field labeled "Nombre cliente" and two buttons at the bottom: "Añadir" and "Cancelar".

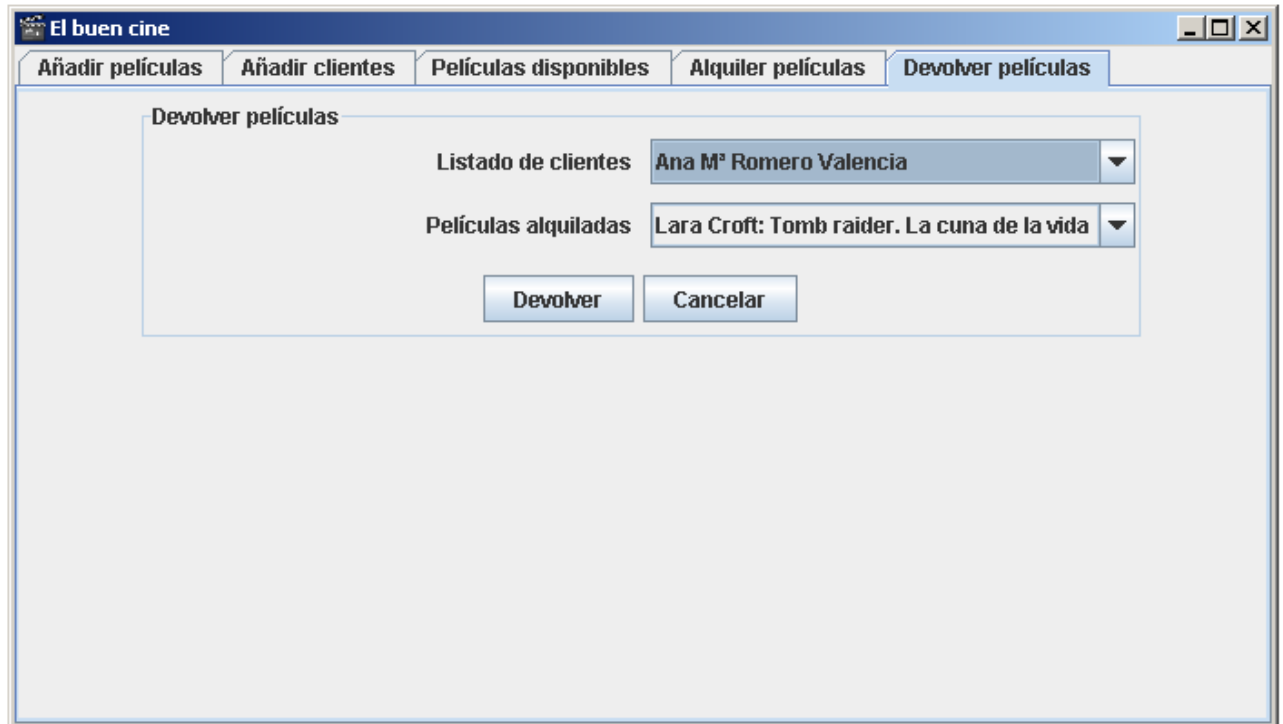
La siguiente pestaña muestra las películas disponibles en el DVDClub, tiene un botón actualizar que nos permite ver las películas que realmente hay, ya que se pueden añadir nuevas, o bien alquilar alguna de las existentes. En el caso de esta pestaña, se puede optar por no poner el botón de Actualizar y que siempre que se entre en el método actionPerformed, se actualice el JTextArea, pero esto lo haría más ineficiente.



La pestaña de Alquiler de películas nos permite que un cliente alquile una o varias películas, como se puede ver se muestran dos JComboBox, uno con los clientes y otro con las películas, en el caso de la lista de películas, según se van alquilando, van desapareciendo del combo. Para que un cliente pueda alquilar una película, tan sólo hay que elegir al cliente, a la película y pulsar el botón Alquilar. Si por el contrario se pulsa el botón Cancelar se mostrará en los combos, el primero de ambas listas.



Por último la pestaña de devolver películas tiene un funcionamiento parecido al caso anterior. En este caso, cuando se selecciona un cliente, en el combo de las películas sólo aparecerán las películas que ese cliente tiene alquiladas, en caso de que tenga alguna. Al igual que antes para que un cliente pueda devolver una película, basta con seleccionar el cliente, la película a devolver y pulsar el botón devolver, en ese momento, la película devuelta debe ya desaparecer del combo de películas. Si se pulsa el botón cancelar se mostrará en los combos, el primero de ambas listas.



Notas:

- Ser claros programando (estructurar bien las clases, poner comentarios, nombres significativos, ...)
- Comprobar que la aplicación funciona correctamente cuando se crea el DVDClub en blanco.

Por último mencionar que todo lo expuesto anteriormente es, por supuesto, mejorable, en cuanto al aspecto del interfaz gráfico, así como al planteamiento y funcionamiento de la aplicación. Por eso, se permite al alumno que modifique todo aquello que estime oportuno siempre y cuando lo justifique adecuadamente.

Posibles mejoras futuras:

- *Mejorar el interfaz gráfico.*
- *Volcar al salir de la aplicación los datos de nuevo en un fichero.*
- *Trabajar con bases de datos en lugar de con ficheros.*
- *Permitir borrar tanto películas como clientes del DVDClub.*
- *Convertir la aplicación en un Applet.*
- *Etc.*