

TAD Lista

Práctica 5 Aplicación de Listas Ordenadas

Objetivo

El objetivo de esta práctica es que el alumno ponga en práctica el uso de listas, en particular listas ordenadas, a través del desarrollo de una aplicación que utilice dicha estructura de datos. Además el alumno reforzará sus conocimientos sobre el almacenamiento y recuperación de objetos persistentes mediante la aplicación del concepto de serialización de objetos de Java.

Descripción general

En un sitio web se desea mostrar una lista de títulos de películas junto con su calificación en orden descendente, cuyo valor puede ser un entero del rango 1 a 10 para indicar qué tan recomendable es la película, siendo 10 la mejor calificación posible.

El programa debe ser controlado a través de un menú de opciones que permita crear una lista de películas ordenada por calificación y cuente con las siguientes operaciones:

- Agregar una película
- Agregar películas desde un archivo
- Mostrar todas las películas
- Consultar una película
- Actualizar la calificación de una película
- Eliminar una película
- Borrar la lista de películas

Este programa deberá permitir que los datos de las películas permanezcan más allá de la ejecución del programa, por lo que hará uso del concepto de serialización para almacenar y recuperar a partir de un archivo en disco los datos de las películas.

Material

El material de esta práctica consta de los siguientes archivos:

- **InterfazLista.class** interfaz para el tipo abstracto de datos Lista.
- **Nodo.class** clase que implementa los nodos que sirven para crear una lista.
- **Lista.class** clase con una posible implementación de la interfaz `InterfazLista`.

- **Lista\$MiIterador.class** clase interna de la clase Lista que implementa un iterador de la lista.
- **ListaOrdenada.class** clase que implementa una lista ordenada. **Nota:** Se usará la clase desarrollada en la práctica anterior.
- **Pelicula.java** clase que se desarrollará para definir la estructura de una película e implementará la interfaz Serializable de Java para que los objetos de la clase puedan serializarse.
- **ListaPelículas.java** clase que creará la lista ordenada de películas así como las operaciones que se podrán realizar sobre ella. También implementará las funciones para serializar y deserializar los objetos de tipo Pelicula y así asegurar la persistencia de los datos.
- **PruebaListaPelículas.class** programa para probar la clase ListaPelículas.
- **datosPelículas.txt** archivo usado en el programa PruebaListaPelículas.
- Documentación:
 - InterfazLista.html documentación de la interfaz InterfazLista.
 - Nodo.html documentación de la clase Nodo.
 - Lista.html documentación de la clase Lista.
 - Serializable.html documentación de la interfaz Serializable de Java.

Desarrollo

1. Descargar los archivos `InterfazLista.class`, `Nodo.class`, `Lista.class`, `Lista$MiIterador.class`, `PruebaListaPelículas.class`, `Pelicula.java` y `ListaPelículas.java` en el directorio donde se va a desarrollar la práctica. **No olvidar** incluir el archivo `ListaOrdenada.class` desarrollado en la práctica anterior.
2. Leer la documentación y revisar el material de la práctica. Recordar que la firma de los métodos proporcionados y de los requeridos en el desarrollo de la práctica no podrán ser modificados.
3. Desarrollar la clase `Pelicula` la cual debe implementar la interfaz de Java `Serializable` además de contar con lo siguiente:
 - a. **Constructor** que reciba todos los atributos.
 - b. **Constructor** que reciba el título de la película y asigna al atributo `calificacion` el valor 1 por omisión.
 - c. Métodos **modificadores**.
 - d. Métodos de **acceso**.
 - e. Método **equals** para determinar que dos películas son iguales siempre y cuando tengan el mismo título sin importar mayúsculas o minúsculas.
 - f. Método **toString** para crear una cadena con el título de la película y su calificación.

4. Escribir la clase `ComparaCalificacion` que implemente la interfaz `Comparator` de Java. Determinará la relación de orden entre dos películas de acuerdo a sus calificaciones.
5. Implementar la clase `ListaPeliculas` para crear la aplicación que permita la creación y manejo de la lista de películas. Se deberán implementar los siguientes métodos:
 - a. **Constructor** para crear la lista ordenada donde se almacenarán las películas. Recibe como parámetro una cadena con el nombre del archivo en disco en el que se almacenarán y recuperarán los datos de las películas. Si el archivo especificado existe, se llevará a cabo la función de deserialización de objetos. Si no existe el archivo, la lista ordenada estará vacía inicialmente.
 - b. Método **leerPeliculas()** que realiza el proceso para deserializar el registro de películas, es decir, recuperar los datos almacenados en el archivo en disco cuyo nombre fue indicado en el constructor y los deposita en la lista ordenada de películas.
 - c. Método **guardarPeliculas()** para serializar los objetos de tipo `Pelicula` que se tienen en la lista ordenada de películas. Cuando el uso del programa termina, se almacenan las películas de la lista ordenada en el archivo en disco cuyo nombre fue indicado en el constructor.
 - d. Método **agregarPelicula(String titulo, int calif)** que insertará en la lista ordenada los datos de una película e imprimirá en pantalla que la película fue agregada. No se permiten títulos de películas duplicados. Si la película ya existe en la lista ordenada, sólo se actualizará el dato correspondiente a su calificación.
 - e. Método **agregarArchivoPeliculas(String nombreArchivo)** que recibe como parámetro el nombre de un archivo de texto con datos de películas. Insertará cada película leída del archivo en la lista ordenada. El archivo contendrá los datos de una película por renglón usando el siguiente formato:
`titulo,calificacion`. No se permiten títulos de películas duplicados. Si la película ya existe en la lista ordenada, sólo se actualizará el dato correspondiente a su calificación.
 - f. Método **imprimirLista()** para mostrar en pantalla la información de las películas contenidas en la lista ordenada.
 - g. Método **consultarPelicula(String titulo)**. Si el título de la película se encuentra en la lista, se mostrará su información. En caso contrario, se mostrará en pantalla que la película no fue encontrada.
 - h. Método **modificarCalificacion(String titulo, int calif)**. Si el título de la película se encuentra en la lista, se modificará su rating e imprimirá en pantalla que la modificación fue realizada. En caso contrario, se mostrará en pantalla que la película no fue encontrada.

- i. Método **eliminarPelicula(String titulo)** Si el título de la película se encuentra en la lista, se eliminará de la lista. En otro caso, se mostrará en pantalla que no pudo ser eliminada pues no se encontró.
- j. Método **borrarLista(String confirmacion)** que recibe la confirmación del usuario para borrar o no la lista de películas. Mostrará en pantalla un mensaje para indicar si la lista fue borrada o no, de acuerdo al parámetro recibido.

Se deberá mostrar al usuario un menú con las opciones disponibles, leer la opción elegida por el usuario y realizar la acción correspondiente a la opción elegida.

Al empezar a trabajar con el programa, este recibirá el nombre del archivo donde están almacenados las películas, en caso de no recibirlo se asume que el archivo será *"misPeliculas.ser"*. Al finalizar el trabajo con el programa, este grabará todas las películas de la lista ordenada dentro del archivo indicado al inicio, o en su defecto en el archivo por omisión *"misPeliculas.ser"*.

El programa debe estar preparado para manejar las posibles excepciones que puedan dispararse al trabajar con archivos y cuando el usuario introduzca parámetros que no estén definidos.

El siguiente es un ejemplo de ejecución del programa solicitado:

```
LISTA DE PELICULAS
1. Agregar una pelicula
2. Agregar peliculas desde un archivo
3. Mostrar todas las peliculas
4. Consultar una pelicula
5. Actualizar calificacion
6. Eliminar pelicula
7. Borrar lista
0. Terminar

Elige una opcion
1

Introduce el titulo de la pelicula
Up
Introduce la calificacion de la pelicula
8

La pelicula ha sido agregada.

LISTA DE PELICULAS
1. Agregar una pelicula
```

2. Agregar peliculas desde un archivo
3. Mostrar todas las peliculas
4. Consultar una pelicula
5. Actualizar calificacion
6. Eliminar pelicula
7. Borrar lista
0. Terminar

Elige una opcion

1

Introduce el titulo de la pelicula

The Shining

Introduce la calificacion de la pelicula

10

La pelicula ha sido agregada.

LISTA DE PELICULAS

1. Agregar una pelicula
2. Agregar peliculas desde un archivo
3. Mostrar todas las peliculas
4. Consultar una pelicula
5. Actualizar calificacion
6. Eliminar pelicula
7. Borrar lista
0. Terminar

Elige una opcion

3

PELICULAS

	Titulo	Calificacion
1.	The Shining	10
2.	Up	8

LISTA DE PELICULAS

1. Agregar una pelicula
2. Agregar peliculas desde un archivo
3. Mostrar todas las peliculas
4. Consultar una pelicula
5. Actualizar calificacion
6. Eliminar pelicula
7. Borrar lista
0. Terminar

Elige una opcion

6

Introduce el titulo de la pelicula por eliminar
the shining

Pelicula eliminada

LISTA DE PELICULAS

1. Agregar una pelicula
2. Agregar peliculas desde un archivo
3. Mostrar todas las peliculas
4. Consultar una pelicula
5. Actualizar calificacion
6. Eliminar pelicula
7. Borrar lista
0. Terminar

Elige una opcion

3

PELICULAS

Titulo	Calificacion
1. Up	8

LISTA DE PELICULAS

1. Agregar una pelicula
2. Agregar peliculas desde un archivo
3. Mostrar todas las peliculas
4. Consultar una pelicula
5. Actualizar calificacion
6. Eliminar pelicula
7. Borrar lista
0. Terminar

Elige una opcion

0

¡Hasta luego!

6. Ejecutar la clase de prueba `PruebaListaPeliculas` para probar los métodos implementados en la clase `ListaPeliculas`. Si el programa funciona adecuadamente se verán los siguientes mensajes:

---- Agregar las siguientes 3 peliculas:

The Godfather,10 Back to the Future,8 Amelie,9

La pelicula ha sido agregada.

La pelicula ha sido agregada.

La pelicula ha sido agregada.

Imprimir lista de peliculas...

PELICULAS		
Titulo	Calificacion	
1. The Godfather	10	
2. Amelie	9	
3. Back to the Future	8	

---- Agregar mas peliculas a partir de un archivo...

Se cargo el archivo de peliculas.

Imprimir lista de peliculas...

PELICULAS		
Titulo	Calificacion	
1. The Godfather	10	
2. The Matrix	10	
3. The Good the Bad and the Ugly	9	
4. Midnight in Paris	9	
5. Back to the Future	8	
6. Jurassic Park	8	
7. 28 Days Later	8	
8. Amelie	8	
9. The Revenant	7	
10. The Intern	4	
11. Jurassic World	3	

---- Eliminar la pelicula 28 days later...

Pelicula eliminada

---- Eliminar la pelicula The Good the Bad and the Ugly...

Pelicula eliminada

Imprimir nuevamente lista de peliculas...

PELICULAS

Titulo	Calificacion
1. The Godfather	10
2. The Matrix	10
3. Midnight in Paris	9
4. Back to the Future	8
5. Jurassic Park	8
6. Amelie	8
7. The Revenant	7
8. The Intern	4
9. Jurassic World	3

---- Consultar la pelicula the matrix...

Titulo	Calificacion
The Matrix	10

---- Actualizar la pelicula The Matrix con calificacion 9...

Calificacion actualizada

Consultar nuevamente la pelicula The matrix...

Titulo	Calificacion
The Matrix	9

---- Almacenando peliculas en archivo en disco...

---- Prueba para asegurarse de que los datos de la lista fueron serializados...

Obtener lista de peliculas...

PELICULAS

Titulo	Calificacion
1. The Godfather	10
2. Midnight in Paris	9
3. The Matrix	9
4. Back to the Future	8
5. Jurassic Park	8
6. Amelie	8
7. The Revenant	7
8. The Intern	4
9. Jurassic World	3

Borrando la lista completa...

Lista borrada