Take Assessment: Practical Quiz 8

Please answer the following question(s).
If the assessment includes multiple-choice questions, click the "Submit Answers" button when you have completed those questions.

You have 120 minutes to take this assessment.
Please complete this assessment by Thu Dec 20 2007 22:34:04 GMT+0800.

1. Go to bottom of question.

VHS and DVD Movies

Prerequisites, Goals, and Outcomes

Prerequisites: Before you begin this exercise, you need mastery of the following:

- *Object Oriented Programming*
  - Knowledge of abstract classes
    - How to define an abstract class
  - Knowledge of interfaces
    - How to define an interface
    - How to define a class that implements an interface

Goals: Reinforce your ability to use Java interfaces and abstract classes

Outcomes: You will demonstrate mastery of the following:

- Writing interfaces
- Writing classes that implement interfaces
- Writing abstract classes

Background

In this assignment, you will create the following classes and interfaces:

- Abstract class
    - Movie
- Interfaces
    - VHS
    - DVD
- Classes
    - VHSMovie
    - DVDMovie

Description

The following class diagram illustrates the relationships between the interfaces and classes:

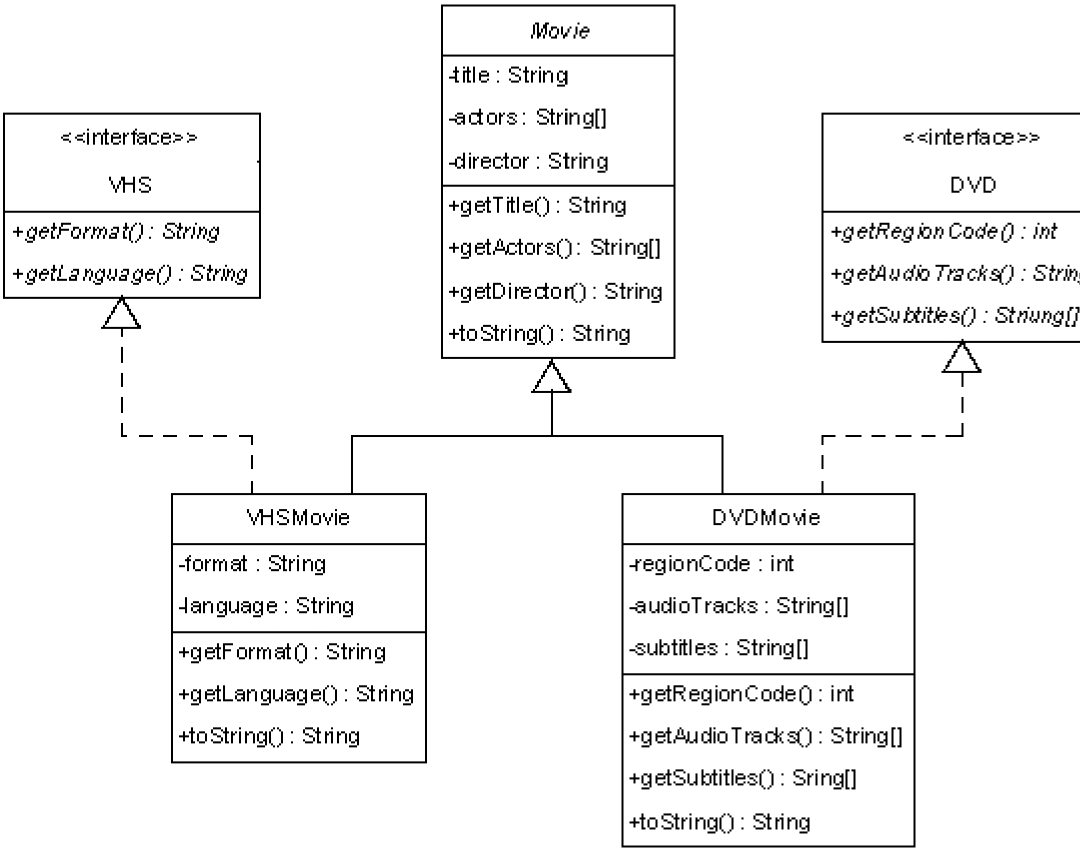

Figure 1 *Class diagram*

The specifications of the interfaces and classes are as follows:

Abstract class Movie

The abstract class Movie stores the information of a movie.

Instance variables:

- *String title*. The title of the movie
- *String[] actors*. The names of the actors in the movie
- *String director*. The director of the movie

Constructor and methods:

- *Movie(String initialTitle,*
- *String[] initialActors,*
- *String initialDirector)*

  Creates a Movie object and initializes the instance variables.

- *String getTitle()*. Returns the value of the variable title.

- *String[] getActors()*. Returns a reference to the array actors.

- *String getDirector()*. Returns the value of the variable director.

- *String toString()*. Returns the value of the variable title.

Interface VHS

The interface VHS declares the methods for obtaining VHS tape information.

Methods:

- *String getFormat()*. Returns the format of the VHS tape.

- *String getLanguage()*. Returns the language of the VHS tape.

Interface DVD

The interface DVD declares the methods for obtaining DVD information.

Methods:

- *int getRegionCode()*. Returns the region code of the DVD.

- *String[] getAudioTracks()*. Returns an array with the names of the audio tracks on the DVD.

- *String[] getSubtitles()*. Returns an array with the languages of the subtitles on the DVD.

Class VHSMovie

The class VHSMovie extends class Movie and implements the interface VHS.

Instance variables:

- *String format*. The format of the VHS movie
- *String language*. The language of the VHS movie

Constructor and methods:

- *VHSMovie(String initialTitle,*
- *String[] initialActors,*
- *String initialDirector,*
- *String initialFormat,*
- *String initialLanguage)*

  Creates a VHSMovie object and initializes the instance variables.

- *String getFormat()*. Returns the value of the variable format.

- *String getLanguage()*. Returns the value of the variable language.

- *String toString()*. Returns a string representation of the object with the following format:

  title, *format*, *language*

  where:

  - *title* is the title of the VHS movie.
  - *format* is the format of the VHS movie.
  - *language* is the language of the VHS movie.

  The fields are delimited by a comma (,). You can assume that the fields themselves do not contain any commas.

Class DVDMovie

The class DVDMovie extends class Movie and implements the interface DVD.

Instance variables:

- *int regionCode*. The region code of the DVD movie
- *String[] audioTracks*. The names of the audio tracks on the DVD movie

- *String[] subtitles*. The languages of the subtitles on the DVD movie

Constructor and methods:

- *DVDMovie(String initialTitle,*
- *String[] initialActors,*
- *String initialDirector,*
- *int initialRegionCode,*
- *String[] initialAudioTracks,*
- *String[] initialSubtitles)*

  Creates a DVDMovie object and initializes the instance variables.

- *int getRegionCode()*. Returns the value of the variable regionCode.

- *String[] getAudioTracks()*. Returns a reference to the array audioTracks.

- *String[] getSubtitles()*. Returns a reference to the array subtitles.

- *String toString()*. Returns a string representation of the object with the following format:

  title, *regionCode*

  where:

  - *title* is the title of the DVD movie.
  - *regionCode* is the region code of the DVD movie.

  The fields are delimited by a comma (,). You can assume that the fields themselves do not contain any commas.

Test driver classes

Complete implementations of the following test drivers are provided in the student archive:

- Class TestMovie
- Class TestVHS
- Class TestDVD
- Class TestVHSMovie

- Class TestDVDMovie

Files

The following files are needed to complete this assignment:

- *student-files.zip* — Download this file. This archive contains the following test drivers:
  - *TestMovie.java*
  - *TestVHS.java*
  - *TestDVD.java*
  - *TestVHSMovie.java*
  - *TestVHSMovie.java*

Tasks

Implement the abstract class Movie, the interfaces VHS and DVD, and the concrete classes VHSMovie and DVDMovie. Document using Javadoc and follow Sun's code conventions. The following steps will guide you through this assignment. Work incrementally and test each increment. Save often.

1. **Extract** the files by issuing the following command at the command prompt:

   ```
   C:\>unzip student-files.zip
   ```

2. **Then**, implement the class Movie from scratch. Use TestMovie to test your implementation.

3. **Next**, implement the interface VHS from scratch. Use TestVHS to test your implementation.

4. **Then**, implement the interface DVD from scratch. Use TestDVD to test your implementation.

5. **Next**, implement the class VHSMovie from scratch. Use TestVHSMovie to test your implementation.

6. **Finally**, implement the class DVDMovie from scratch. Use TestDVDMovie to test your implementation.

Submission

Upon completion, submit **only** the following:

1. Movie.java
2. VHS.java
3. DVD.java
4. VHSMovie.java
5. DVDMovie.java

Go to top of question.

File to submit: Upload File Forward File Refresh Ready for Grading

Go to top of assessment.