

CSCI 2540 Assignment 6

100 points

Due date: Tuesday, March 14 (before class)

In this assignment, you will design and create an **ArrayList**-based application that manages a collection of DVDs. Information about a DVD includes title, category, running time, year, and price. The list of DVDs and the information for each DVD are stored in a text file before and after the program is executed.

You will need to write the code for one interface (**DVDCollectionInterface**) and three classes (**DVD**, **ArrayListDVDCollection**, and **DVDApplication**), for which the details are given below.

Interface

- **DVDCollectionInterface** - The interface file for the DVD collection. This interface should include the following methods:
 - *loadData* – this method should have a file name as the parameter. The method loads the data containing the DVD collection entries from a given file.
 - *lookupEntry* – this method should have a DVD title as the parameter. It should return the DVD entry if found, or null if not found.
 - *addOrChangeDVD* – this method is used to add a new DVD or change the information of an existing DVD entry. It should have five parameters that represent the title, category, running time, year, and price of a DVD. If the title is not currently in the DVD collection, then add the DVD entry to the collection. Otherwise, if the title is already in the collection, modify the information of the existing DVD entry accordingly. Return the DVD entry if it is changed, or return null if the new DVD is added.
 - *removeDVD* – this method should have a title as the parameter. It should remove the DVD from the collection if the title is found. It returns the DVD entry which was removed, or null if the title is not found.
 - *getDVDsInCategory* – this method should have a category as the parameter. It should return an arrayList object with all the DVDs in the specified category. If there is no DVD in the given category, the size of the arrayList will be zero.
 - *Save* – this method has no parameter. It should write the DVD collection back to the file (the same file is used for both reading and writing).

Please include comments for all the methods in the interface.

Classes

- **DVDApplication** - The class for the main program. You will need to read the data from input file, display menu options, and perform various tasks that user selects.
- **ArrayListDVDCollection** - The class for the actual DVD collection. The DVD collection must be stored in an **ArrayList** object. You will need to write the code for all the methods specified in the **DVDCollectionInterface**.

You may add additional methods as needed to this class, however ALL additional methods you add MUST be declared as **private**.

- **DVD** - The class containing all of the information about one DVD.
 - The information to be stores is: the title of the DVD, the category for the DVD, the running time for the DVD, the year the DVD was released, and the price for the DVD.
 - You should include getter methods for ALL five of the data fields.
 - You should include setter methods for all of the data fields EXCEPT title. The title of the DVD is not allowed to be changed.
 - You will need to write the code for the *toString* method.
 - You will need to write the code for the *equals* method. This method should have one parameter of Object type. Two DVD objects are considered to be equal if they have the same titles.
 - You will need to write the code for the *compareTo* method. This method should have one parameter of DVD type. This method is to test the order of two DVD objects to see which one comes first alphabetically by their titles. Note that in order to have this method, your DVD class should implement the Comparable<DVD> interface.

Program Requirements

- As your program begins, it should read previously saved data from a file. The name of the file will be **DVDCollection.txt**. The data in the file should be in the following format; i.e. each piece of data should be on a separate line - title on the first line, category on the second line, running time on the third line, year of release on the fourth line and price on the fifth line. The input file contains information for multiple DVDs in the above format. A sample input file is posted on Blackboard. (While you can use this sample input file to test your program, the instructor/grader may not necessarily use it to test your programs.)

- The program should then display a menu of options to the user and allow the user to make choices from the menu. This continues until the user selects the exit option from the menu. Upon exiting, the program should save the DVD collection back to the file **if any of it has been modified**.

Menu Options

- **Add new DVD entry** - the user should be able to add new DVDs to the collection. First ask user to enter the title. Check whether the title is already in the collection. If so, display a message and return to the main menu. Otherwise, ask user to enter the rest four pieces of information associated with a DVD, and add the new DVD to the collection.
- **Change existing DVD entry** - the user should be able to change the information for a DVD that is already stored in the collection. The procedure for changing data is as follows:
 - Ask the user to type in a DVD title
 - Search for the title in the collection.
 - If the title is not found, display a message to the user stating that item is not in the collection and then return to the main menu to allow them to make another choice.
 - If the title is found, retrieve all of the information for the requested DVD and display a submenu on the screen so the user can choose which item he/she would like to change:
 - 1: Title: *show the title of the DVD*
 - 2: Category: *show the category for the DVD*
 - 3: Running Time: *show the running time for the DVD*
 - 4: Year of Release: *show the year of release for the DVD*
 - 5: Price: *show the price of the DVD*
 - 0: Return to main menu
 - The user should be allowed to choose from this menu, making changes to any of the DVD's data fields (except for title) until they enter option 0, which should return them back to the main menu. If the user chooses 1 for the submenu attempting to change title, the program should print a message that title cannot be changed.
- **Look Up DVD Entry** - the user should be allowed to look up the data for a DVD in the collection. Ask the user to enter a DVD title. If the title is not in the collection, display a message to the user stating that the item is not in the collection. If the title is found, display all of the data associated with the requested title - one item per line. Make sure to label each item as it is displayed.
- **Display DVDs By Category** - the user should be able to display all DVDs in a particular category. Ask the user to enter a DVD category. If there are no DVDs in the collection matching the requested category, display a message to the user stating that there are no DVDs in the requested category. Otherwise, display a list of DVD titles - one title per

line - for the DVDs in the specified category. Display only the titles, and not all of the information.

- **Remove DVD Entry** - the user should be allowed to remove a DVD from the collection. Ask the user to enter a DVD title. If the title is not in the collection, display a message to the user stating that the item is not in the collection. If the title is found, display all of the data associated with the requested title - one item per line - along with a message that the DVD has been removed from the collection.
- **Save Data** - the user should be allowed to save the DVD collection back to the file. The save option should not actually do anything if the collection has not been modified.
- **Exit** – exit the program. Upon exiting, the program should save the DVD collection back to the file if any of it has been modified.

Submission instructions:

To submit your programs, you need to submit your programs electronically on Blackboard. Please also bring a hard copy of your programs to the class to submit.

Make sure you put all your source code files (and only source code files) in one package and name the package as assg6_Yourlastname. When you submit your files to Blackboard, please submit your package folder (with source code only, i.e., .java files) as one zip file.