

CS 172 – Homework 2

Purpose:

After completing this assignment, you will have practiced overloading operators for a programmer-defined class and using inheritance.

Description

For this assignment you will write a program to simulate a media library. We will work with three types of media: pictures, music, and movies. Your job is to design and implement a class hierarchy consisting of the `Media`, `Picture`, `Song`, and `Movie` classes, as described below.

Specification for the classes:

Media class

A `Media` object has a media type, a name, and a rating. The rating is an integer number. You need to implement the `__init__`, `getter`, and `setter` methods as needed.

This class should overload the `__str__` or `__repr__` methods so that `Media` objects can be displayed using the `print()` function.

Movie class

A `Movie` is a type of `Media` that has a director and running time (given in minutes). This class has a `play()` method that will simulate the task of playing a movie (you can do something simple as printing: `'<<movieName here>>, playing now'`). The `Movie` class should override the `__str__` (or `__repr__`) method of `Media`. Make sure you also implement any other methods (such as `__init__`, `getters`, and `setters`) as needed by `Movie`. You should be able to use the `__str__` (or `__repr__`) method in a script to show all the movie information, including media type, name, rating, director, and running time.

Song class

A `Song` is a kind of `Media` that has an artist and an album. This class has a `play()` method that will simulate the task of playing a song (you can do something simple as printing something like: `'<<SongName here>> by <<artistName here>>, playing now'`). The `Song` class should override the `__str__` (or `__repr__`) method of `Media`. Make sure you also implement any other methods (such as `__init__`, `getters`, and `setters`) as needed by `Song`. You should be able to use the `__str__` (or `__repr__`) method in a script to show all the song information, including media type, name, rating, artist, and album.

Picture class

A `Picture` is a kind of `Media` that has a resolution. The resolution of a picture is an integer number that measures the dots per inch (the minimum resolution of any picture should be 200 dpi). This class has a `show()` method that will simulate the task of displaying a picture (you can do something simple as printing: `'Showing <<pictureName.extension>>'`). The `Picture` class should override the `__str__` (or `__repr__`) method of `Media`. Make sure you also implement any other methods (such as `__init__`, getters, and setters) as needed by `Picture`. You should be able to use the `__str__` (or `__repr__`) method in a script to show all the picture information, including media type, name, rating, and resolution.

Script

Your script will simulate a media library. You should have a list of `Media` that stores at least 12 different objects. Your list should have a mix of `Song`, `Movie`, and `Picture` objects. You will provide the information for each media object when you populate the list.

Your program should have a menu that allows the user to perform the following actions:

- Display all items in the `Media` library
- Display only the `Song` objects
- Display only the `Movie` objects
- Display only the `Picture` objects
- Play a `Song`: the user enters the name of the `Song`. If the `Song` is found play it. If not, display a message indicating that the `Song` is not in the media library.
- Play a `Movie`: the user enters the name of the `Movie`. If the `Movie` is found play it. If not, display a message indicating that the `Movie` is not in the media library.
- Display a `Picture`: the user enters the `Picture`. If the `Picture` is found display it. If not, display a message indicating that the `Picture` is not in the media library.
- Quit the program

You will need a loop to show and process the menu until the user chooses to quit/exit the program.

NOTE: Please keep in mind that you are expected to write a good quality, well formatted program. That means:

- Your program must have a header comment listing your full name, Drexel user id, and the purpose of the file – at the very list.
- User input must be validated and your program gracefully handle invalid inputs.
- Repetitive code (code that appears in multiple places in the main script) should be written as a function.
- Your program must use good style, including proper identifier names, useful comments, and proper use of indentation and whitespace.
- Your program should also have an appropriate user interface so that anyone one using the program knows what to do and what to expect.

Grading

Criteria	Points
Media class: <code>__init__</code> getter and setter methods	12
Media class: either <code>__str__</code> or <code>__repr__</code> overloaded method	10
Movie/Song/Picture classes: <code>__init__</code> getter and setter methods	12
Movie/Song classes: <code>play()</code> method & Picture class <code>show()</code> method	12
Movie Movie/Song/Picture classes: <code>__str__</code> or <code>__repr__</code> overridden method	12
Main script: properly creates at least 12 media objects of different types and stores them in a list	6
Main script: validates user's input	6
Main script: all the required parts are there: menu options are handled correctly	20
Code follows good style guidelines and separate repetitive code into functions	10
Total possible points	100

NOTE: If you code has any runtime errors a 50-point deduction will be taken. Only portions of the code that execute without errors will be graded. If your script cannot run at all, you will receive 0 points.

How to Submit your assignment:

- Assignments must be submitted via Blackboard Learn.
 - Please note that assignments submitted via email will not be accepted.
 - Late assignments will lose 1% per hour up to 48hrs (after which they will not be accepted).
- For this assignment, you must submit a single zip (such as HW2.zip) file that contains:
 - **media.py** – file that contains all the classes required for this assignment.
 - **main.py** - your main script

Academic Honesty

You must be the **sole original author** of the **entire solution** you submit. You must compose all program and written material yourself. All material taken from outside sources (e.g. textbooks, in class examples, labs, etc.) must be appropriately cited.