



COMP231 – Advanced Programming

Assignment # 3

Objectives:

1. Create hierarchy of Classes and Objects using Inheritance relationships.
2. Demonstrate the added value of using the following concepts:
 - Polymorphism, Generic binding, Inheritance, Abstract classes and Interfaces

Specification

Submission: **Online through Ritaj.**

What to submit: Your **OWN** well-structured and well-commented JAVA files (.java) (compressed into a studentId_assignment3.rar file, e.g. **1234567_assignment3.rar**).

Deadline: **17/8/2017** by midnight. (The online submission will be disabled after this time).

Task 1:

Write a **PrintCalendar** class to display a calendar for a specified month using the **Calendar** and **GregorianCalendar** classes. Your program receives the month and year from the command line. For example:

java PrintCalendar 5 2016

This displays the calendar shown in the following:

```
Command Prompt
c:\exercise>java PrintCalendar 5 2016
May, 2016
-----
Sun Mon Tue Wed Thu Fri Sat
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31
c:\exercise>
```

You also can run the program without the year. In this case, the year is the current year. If you run the program without specifying a month and a year, the month is the current month.

Task 2:

In a movie library application, you are asked to consider the following kinds of movies:

- **Movie**, a class describing all kinds of movies.
- **Action**, a movie containing lots of explosions.
- **Romance**, a movie where romantic interest drives the plot.
- **Comedy**, a movie with largely humorous content.
- **Mystery**, a who done it movie.
- **Rescue**, a hybrid *action-romance* movie, where the main character attempts to save his or her romantic interest from almost certain doom.
- **Romantic Comedy**, a hybrid *romance-comedy* with large amounts of both humorous and romantic content.
- **Hollywood Blockbuster**, an *action-romance-comedy-mystery* movie designed to please crowds.

- What **interfaces** and **classes** would you use to represent the previous list of movies? Write your answer by carefully drawing a **UML class/interface** hierarchy, identifying which nodes are **classes** and which are **interfaces**.
 - Note that there must be a class for each type of the movies, but you may use any interfaces you require to preserve the relationships between types.
- Adequately, implement all the classes/interfaces. Include whatever methods/parameters necessary in each class/interface.
- Your classes should all implement the **Cloneable** and **Comparable** interfaces.
- Force all the sub-classes to override **toString** method.
- Create a text file "**movies.txt**" that includes information about variety of movies. The following is a sample **movies.txt** file:

Type: Title, [Starring], Running time, Country, Language.

Comedy: Minions, [Sandra Bullock, Jon Hamm], 91, USA, English.

Action: The Matrix, [Keanu Reeves, Laurence Fishburne], 120, USA, English.

Hollywood Blockbuster: Gladiator, [Russell Crowe, Joaquin Phoenix], 155, USA, English.

Mystery: Harry Potter, [Daniel Radcliffe, Rupert Grint, Emma Watson], 178, UK, English.

Action: Entrapment, [Sean Connery, Catherine Zeta-Jones], 113, USA, English.

Comedy: Dumb and Dumber, [Jim Carrey, Jeff Daniels], 107, USA, English.

Rescue: Avatar, [Sam Worthington, Zoe Saldana], 161, USA, English.

Romance: Titanic, [Leonardo DiCaprio, Kate Winslet, Billy Zane], 195, USA, English.

- Write a test program that do the following:
 - Open and read the **movies.txt** file that you created in the previous step.
 - Convert each line to the appropriate movie class based on the first token (e.g. **Comedy**).
 - Create an **ArrayList** of movies to store the created movies from the previous step.
 - Write a method to sort the **ArrayList** of movies based on movie type and running time.
- Write the sorted **ArrayList** of movies into a file called "**sorted_movies.txt**".

Good Luck!

