COMP2130 Assignment 2

Due: Friday, November 29th 2019

11:00 PM

Group size: 2 to 3 persons (==from your lab session==)

This Assignment is designed to take you through the process of creating basic classes and integrating them with a simple JavaFX generated interface.

Scenario:

A chess club wants your group to develop some software that supports a number of its activities. The software is to be operated by the club secretary who will use it to manage membership details. This will involve adding and removing members from the club as well as updating the number of official games each member has played, their wins and losses.

The system must record for each member, their first name, last name, number of games played, number of wins and number of losses (number of games – wins).

The system should also allow the club secretary to see a list of all the members and their win rate (number of wins divided by the number of games played). It will also be possible for the secretary to add a new win or loss to a member's record.

Your group must create a program that simulates a simple club interface allowing the addition of new members and all the operations outlined above. It also allows the viewing of all accounts in the system. JavaFX must be used to create the interface for the system.

Member services:

1. Add member
2. Record win
3. Record loss

Reporting Services:

1. List Members
2. View Best Player (The player with the highest win rate, If there are ties show all players that are the best)
3. View player with the most wins (note this may be different from the Best Player, If there are ties show all players with the most wins)

Overview:

A class MemberManager class must be created that manages all of the members in the system. A single instance of the MemberManager class must be created for this application. The MemberManager object may be declared in the main javafx application class and instantiated in the init( ) method of the JavaFX application.

Marking Scheme:

Attribute	Marks Allotment
JavaFX interface completion	30
Core OOP Design and implementation	50
Coding Standards (indentation, classes capitalized, etc.)	10
Integration of all parts of the system	10
TOTAL	**100**

Submission Requirements:

- Submission: On Blackboard.
- Required files:
    - o Your exported project must also be compressed and uploaded as a .zip file.
    - o In addition, copy each Java class code into one Word document (.doc file). Add comments for the filename at the top of each Java class source code. Submit this .doc file also on Blackboard.
- This project is a group project.
- The full names and student numbers of all group members must be at the top of each class.
- Every member of the group must upload the Assignment 2 required files to Blackboard.

CAREFUL NOTE:

- Please safeguard your own code work.

- If 2 or more assignments are the same (or very much alike) they will all get 0 marks, so be cautious not to share your application with others.

- Note: Mobi-Help members are not supposed to do or help you with your code for the assignments.