CSIT121/CSIT821 - Object Oriented Design and Programming

Assignment 3 (due in week 12 & 13)

The purpose of this assignment is to practice file IO and HashMaps. This assignment continues the scenario from assignment 1 and 2. This assignment contains three levels: core, standard and advanced. They should be done in this order. The core level does not require a swing GUI and so can be done by only using my assignment 1 solution code. If you do not have a reasonable working assignment 2 you should focus on completing assignment 3 at the core level.

Core Level
The user will be able to see the following overviews:

Card overview
- Number of cards
- Total points for all cards
- Total balance for all cards

Purchase overview
- Number of purchase made
- Total value of all purchases
- Total value of all purchases for each of the five categories

The program will load prior cards and purchases into internal lists when its starts up. The program will save the lists of cards and purchases (including any additional ones) when the program is shutdown.

If you are only doing the core level, then you may use a console interface i.e. simple command line. In this case, the user will be able to issue a shutdown command from the console. Also if you are only doing the core level, the user must still be able to create additional cards and make additional purchases, which may be done via the console. Follow the recommended steps on the next page.

Standard Level
A standard level submission includes all the requirements for the core level, except you are required to use a swing GUI. The program will provide a button or menu for the shutdown operation. The program will also maintain a log file (a text file) of all card and purchases transactions. Each transaction should be recorded in the log file as a single line in some delimited format and include a time stamp.

The use will be able to create additional cards and purchases via the swing GUI (from assignment 2). Such additional cards and purchases will be stored into their respective text files when the program is shutdown. In the purchase overview: the calculation of the total value of purchases for each category (the five standard categories) will use a HashMap to do the aggregation (as shown in lecture 9).

Advanced Level
An advanced level submission includes all the requirements for the standard level. The advanced level will provide additional analytics: in particular it will allow the administrator of the program to filter the purchase category totals in a variety of ways.

A separate tab pane (or similar) should be created for the following features. The centre of the pane will be a text area, Jtable or similar. It shows the total purchases for each category. The user will be able to filter the purchase data by date/time in several ways. The user will be able to set a date interval, by setting a start date and an end date. The start date should be prepopulated with the earliest purchase date, and the end date should be prepopulated with the last purchase date. The user will have the option of setting a day-of-the-week slider and a time-of-the-day slider. These sliders will use JSlider. The totals shown in the centre of the pane will incorporate all the filters: the date filter, the day-of-the-week filter and the time-of-the-day filter. The user may use one, two or three sliders at the same time. You will need to use an appropriate Java Date class.

Marking (CSIT121 students)

Advanced (up to 7 marks)

Must be marked in the week 12 lab. Your work must be ready for marking **30 minutes** after the start of the laboratory. You work is likely to be audited, that is, you will be asked to modify and/or explain your code. Your code is expected to be well structured and easy to read.

Standard (up to 5 marks)

Must be marked in the week 12 or 13 lab. Your work must be ready for marking **55 minutes** prior to the end of the laboratory. You work may be audited, that is, you may be asked to modify and/or explain your code. Your code is expected to be well structured and easy to read.

Core (up to 3.5 marks)

Must be marked in the week 13 lab. Your work must be ready for marking **30 minutes** prior to the end of the laboratory. You work may be audited, that is, you may be asked to modify and/or explain your code. Your code is expected to be well structured and easy to read.

Marking (CSIT821 students)

Advanced in week 12 (up to 7 marks)

Your work must be ready for marking **30 minutes** after the start of the laboratory. You work is likely to be audited, that is, you will be asked to modify and/or explain your code. Your code is expected to be well structured and easy to read.

Standard (up to 5 marks)

Must be marked in the week 12 or 13 lab. Your work must be ready for marking **55 minutes** prior to the end of the laboratory. You work may be audited, that is, you may be asked to modify and/or explain your code. Your code is expected to be well structured and easy to read.

You must submit your Java project (zip your project directory) **via the eLearning site** (the assignment 3 drop box) once you have been successfully marked *in the laboratory*.

Recommended steps

1. Design a file format for each of the files.

2.
(a) Create a console netbeans project then write the code needed for reading each of the files your assignment 3 requires using the designed file formats. Write code that loads each file and convert the fields (in each line) into their appropriate format. Then just echo each field to the console for checking.
(b) Then extend the code to create objects, for example, if you are loading item types, extend the code to create item type objects. Then add test code that prints a list of item types (the ones you loaded from the file) to the console.
(c) Further extend your code to support inheritance. If you are loading card types, extend this code so each of the card type subtypes can be loaded i.e. anon, basic and premium cards.

It's only after all this backend code is working and fully tested as standard alone console programs should you consider extending it to interactively add cards and purchases.