BACS2023 OBJECT-ORIENTED PROGRAMMING 2024

ASSIGNMENT SPECIFICATION

Introduction	This is a group assignment. Students are to work in teams of maximum 3-4 members to create an object-oriented program for the given scenario.
Learning Outcomes Being Assessed	CLO2: Develop an object-oriented program using appropriate programming fundamentals with regards to arrays, methods, and exception handling. CLO3: Analyse the concepts of encapsulation, inheritance and polymorphism based on programming problems.
Submission **Deadline**	**12 May 2024 (Sunday) by 11:59 pm** Late submission of the assignment will be handled according to **TAR UMT's** Guideline for **Late Submission of Coursework**. No late assignments will be accepted (**get zero**). Please do not argue with your tutor if you really failed to submit your assignment on time as the consequence of late submission has been given in advance. However, in certain circumstances, the students may be allowed to turn in the assignment late. The students must contact the tutor BEFORE the assignment is due. The tutor will evaluate whether the circumstance warrants submitting the assignment late. A late penalty will be applied. The penalty is as follows: • Late submission within 1 - 3 days total marks to be deducted is 10 marks. • Late submission within 4 - 7 days: total marks to be deducted are 20 marks. • Late submission after 7 days: reject coursework and zero mark shall be awarded.
Overview	**Point Reward system**

Description	In this assignment, students are required to build a professional **Point Reward system** with object-oriented programming (OOP) approach using Java programming language. You are required to develop a simulated environment where output can be a **console output of characters or graphical user interface (GUI)**. However, both display methods are acceptable without any mark discrimination. *"A point reward system is a type of loyalty program used by businesses to encourage customer engagement and loyalty."* It is important to note that all students' team ideas must be **different** and **unique** from other student teams. You can innovate by having different types of reward: E.g. • Discounts and Cashback: Get discounts on future purchases or even redeem them for cashback. • Products or Services: Exchange points for products or services offered by the company. • Exclusive Experiences: Purchase of unique or limited-edition items. • Loyalty Status: Grant access to additional perks and benefits. Your application is represented as Java objects in your system. You need to define the classes for these objects with advanced object-oriented programming features such as **polymorphism**, **inheritance,** and **encapsulation** to control and manage the objects in your system. Also, you are encouraged to use **Java's interface** to establish weak relationships between objects in your system. Each student needs to handle at **least 1 module** in the system. The following are some modules that can be considered to be included in and the students are highly encouraged to do their own research to find more information. • **Contact Management:** This module stores customer and prospect contact information, such as names, addresses and phone numbers. It helps in tracking all interactions with contacts, ensuring that customer information is organized and easily accessible. • **Points Earning**: Define how customers can earn points. • **Points Redemption:** Outline how points can be redeemed. Example discounts on future purchases, free products or services, exclusive access to sales or events, etc. • **Policy**: Decide if points will expire and, if so, the duration before expiration. • **Reporting:** Produce useful reports that help in strategically managing the rewards program, improving customer experience, and ultimately driving business growth. **Do not** use copyrighted images for your display if you are going for GUI display. You can create your own images or use publicly available licensed (e.g., creative common images) images with proper citation of the original source of the image.

<table>
<tr><td>Deliverables</td><td>The following items are to be handed in:

Initial Stage (Dateline – 14 April 2024)
A cover page.
A description of your team's assignment idea.
A description of the modules that are handled by the students.
UML class diagram that depicts the entity classes and their relationships.
Final Report (Dateline – 12 May 2024)
Initial stage report (Students to submit amended version based on the tutor's comments).
Sample screen shots and reports/listings with brief explanation.
Java project source code in the softcopy submission (Google Classroom).
Include ALL your source code files and all pre-compiled classes.
Form of submission: -
Each group creates a folder named using the format <code>TutorialGroup-StudentFullNamesWithAlphabeticalOrder</code>(e.g.,<code>RSF1(S2)-CheahLiMei-HengTzeSeong-NgSiewYongAlice</code>), and to be attached together with the report.
NOTE: Submitting the assignment means you have agreed that your work is original and comply with the rules and regulations (refer to Academic Impropriety)</td></tr>
<tr><td>Paper Size / Submission Format</td><td>Digital Submission (Softcopy)</td></tr>
<tr><td>Estimated Time Required</td><td>At least 10 hours per team member.</td></tr>
</table>

<table>
<tr><td>Academic Impropriety</td><td>You may only work with the students in your team to produce your deliverables for this assignment.

This covers cheating, attempts to cheat, plagiarism, collusion, and any other attempts to gain an unfair advantage in assessment.

The work that you submit must conform to those regulations.

NO-CHEATING POLICY

A reminder on the no-cheating policy: You are NOT to share your work with your peers, but please feel free to have discussion with your peers. If cheating is discovered, both parties will take equal blame (get zero). Please note that the assignment should be your own work, although you may incorporate ideas or techniques from books, online resources, etc. Copying materials directly from any sources of materials will lead to zero. You have been warned. Whenever you face any problems, please seek advice from your tutor.</td></tr>
<tr><td>Assessment</td><td>This assignment contributes 60 marks to your coursework. The allocation of marks is shown below.

Refer to the Assignment Feedback Form for the detailed assessment criteria.

<table><tr><th>Aspect</th><th>Weightage</th></tr><tr><td>Completeness</td><td>15</td></tr><tr><td>Program Output</td><td>5</td></tr><tr><td>Exception Handling</td><td>5</td></tr><tr><td>Implementation of Methods</td><td>5</td></tr><tr><td>Implementation of Array</td><td>5</td></tr><tr><td>Class Design</td><td>30</td></tr><tr><td>Extra features/ functionalities</td><td>5</td></tr><tr><td>Report</td><td>10</td></tr><tr><td>Peer Assessment (Individual)</td><td>20</td></tr></table></td></tr>
<tr><td colspan="2">Note: If it is an individual assignment, the total group mark will be converted to 100%.</td></tr>
</table>