Programming II

Assignment #2 – Classes and Objects

Due Date: **Sunday 17th Feb. 2019 - at the start of the class – 8:30am**

Marks/Weight: **30 / 10%**

Purpose: The purpose of this Assignment is to:

- Practice the use of classes and objects in C#

References: - Text book **"Visual C# 2017, Intro to Object Oriented Programming"**, Chapter 09
- Classes and Objects:
 - https://www.geeksforgeeks.org/c-sharp-class-and-object/
 - https://www.guru99.com/c-sharp-class-object.html
 - https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/classes-and-structs/constructors
- UML:
 - https://www.youtube.com/watch?v=WI0oyCeon2A
 - https://www.lucidchart.com/pages/uml-class-diagram?a=0#top-info
 - https://www.youtube.com/watch?v=WI0oyCeon2A
- Lecture notes/ppts

Instructions: Be sure to read the following general instructions carefully:
This assignment should be completed in groups of 3 students. Submit the project **through e-Centennial, Assessments / Assignment**. You must name your Visual Studio solution according to the following rule: ***GroupCode_COMP123_AssignmentNumber***
For Example: **sec006-6_COMP123_02**

Each exercise should be added to the solution as separate project. Your IDE is Visual Studio 2017 and C#.

Submit your assignment in a **zip file** that is named according to the following rule:
GroupCode_COMP123_AssignmentNumber.zip
Example: **sec006-6_COMP123_02.zip**

Apply the naming conventions for variables, methods, classes, and namespaces:
- *variable names, parameters and fields* – use camelCasing
- *classes, namespaces, methods, properties, enumerations* – use PascalCasing
- *constants:* SNAKE_UPPERCASE

Exercise:

a) *[15 marks]* - Write a C# application using VS 2017 as IDE, that implements the following class as per business requirements mentioned below:

- Create an Employee class (Employee.cs) that has the following UML class diagram:

Employee Class
Fields
- employeeId : int
Properties
+ EmployeeId : int «readonly» + FirstName : string + LastName : string + BaseSalary : double + GrossSales : double + ComissionRate : double
Methods
+ Earnings() : double + «constructor» Employee() + «constructor» Employee(employeeID : int, firstName : string, baseSalary : double)

- Employee ID, first name, last name, base salary, gross sales (amount in dollars) and commission rate. Define their data types appropriately.
- Define read only property for employee ID.
- Use default value of 1000.00 dollars for base salary for all the employees.
- Commission rate should be set by default to 0.1.
- Class should have defined two overloaded constructors:
    - One, without parameters, for initializing all the instance data members
    - Second for initializing employee ID, first name, base salary only.
- Define a public method called earnings which calculates employee's commission (commission rate * gross sales + base salary)

b) *[15 marks]* – Rename Program class to **EmployeeTest**. In the **Main()** method of the EmployeeTest class, create two objects of the type Employee. Each to demonstrate the use of a different constructor that has been created
   - For the second object, ask the user to enter first name, last name, base salary, gross sales and commission rate.
   - After the user enters all information, the application will calculate Earnings and display it formatted as currency.
   - Gross sales and commission rate should never be negative or zero. You need check and validate that the user is entering an acceptable value.
   - Commission rate should be between 0.1% and 1.0%. You need check and validate that the user is entering an acceptable value.

Aspects that will be evaluated:

- Proper identifier casing
- Correct implementation of classes (instance variable declarations, constructors, getter and setter methods)
- Declaring and creating objects, calling their methods, interacting with user, displaying results
- Comments, correct naming of variables, methods, classes
- Friendly input/output