PROG 8010 Assignment Week 9

You are only required to complete the programming problem that has been assigned to your group. However, you are encouraged to work through as many programming problems as possible.

Each group is to submit one solution to eConestoga by Thursday, July 18 before class. Someone from your group will be selected at random to present their solution to the class. Your mark on the assignment will depend on a combination of the quality, functionality, and adhesion to coding standards of your code. If you are absent without excuse, your mark for the presentation portion of the assignment (20%) is zero.

Group 1 Problem – Car Class

Create a class named Car that has the following properties:

- Year – the Year property holds the car's year model.
- Make – the Make property holds the make of the car.
- Speed – the Speed property holds the car's current speed.

In addition, the class should have the following constructor and other methods:

- Constructor – the constructor should accept the car's year and make as arguments. These values should be assigned to the object's Year and Make properties. The constructor should also assign zero to the Speed property.
- Accelerate – the Accelerate method should add 5 to the Speed property's value each time it is called.
- Brake – the Brake method should subtract 5 from the Speed property's value each time it is called.

Demonstrate the class in an application that creates a Car object. The application's form should have an Accelerate button that calls the Accelerate method and then displays the car's current speed each time it is clicked. The application's form should also have a Brake button that calls the Brake method and then displays the car's current speed each time it is clicked.

Group 2/7 Problem – Personal Information Class

Create a class that holds the following personal data in properties: name, address, age, and phone number. Demonstrate the class in an application that creates three instances of the class. Once instance should hold your information, and the other two should hold your friends' or family members' information. Display each object's properties on the application's form.

Group 3 Problem – E-Mail Address Book

Create an application with a class named PersonEntry. The PersonEntry class should have properties for a person's name, email address, and phone number. Also, create a text file that contains the names, email addresses, and phone numbers for at least 5 people. When the application starts, is should read the data from the file and create a PersonEntry object for each person's data. The PersonEntry objects should be added to a List, and each person's name should be displayed in a list box on the application's main form. When the user selects a name from the list box, a second form should appear displaying the person's name, email address, and phone number.

Group 4/8 Problem – Dorm and Meal Plan Calculator

A university has the following dormitories:

Allen Hall	$1,500 per semester
Pike Hall	$1,600 per semester
Farthing Hall	$1,800 per semester
University Suites	$2,500 per semester

The university also offers the following meal plans:

7 meals per week	$ 600 per semester
14 meals per week	$1,200 per semester
Unlimited meals	$1,700 per semester

Create an application with two forms. The main form should allow the user to select a dormitory and meal plan. The application should show the total charges on the second form.

Group 5 Problem – Retail Item Class

Write a class named RetailItem that holds data about an item in a retail store. The class should have the following properties:

- Description – a brief description of the item
- UnitsOnHand – the number of units currently in inventory
- Price – the item's retail price

Write a constructor that accepts arguments for each property.
The application should create an array of three RetailItem objects containing the following data:

Description	Units on Hand	Price
Jacket	12	59.95
Jeans	40	34.95
Shirt	20	24.95

The application should have a loop that steps through the array, displaying each element's properties.

Group 6/9 Problem – Employee Class

Write a class named Employee that has the following properties:

- Name – employee's name
- IdNumber – the employee's ID number
- Department – the name of the department in which the employee works
- Position – the employee's job title

The class should have the following overloaded constructors:

- A constructor that accepts the following values as arguments and assigns them to the appropriate properties: name, ID number, department, and position.
- A constructor that accepts the following values as arguments: name, and ID number. The department and position properties should be assigned an empty string.
- A parameterless constructor that assigns empty strings to the Name, department, and position properties, and a zero to the IdNumber property.

In an application, create three Employee objects, populate them with data and display the data on screen.