C# BASICS

Training Assignments

Document Code	25e-BM/HR/HDCV/FSOFT
Version	1.1
Effective Date	20/11/2012

Hanoi, 06/2019

RECORD OF CHANGES

No	Effective Date	Change Description	Reason	Reviewer	Approver
1.	01/Oct/2018	Create new	Draft		
2.	01/Jun/2019	Update template	Fsoft template	DieuNT1	

Contents

FRESHER ACADEMY	**CODE:**	**NPL.M.A007**
	TYPE:	**MEDIUM**
	LOC:	
	DURATION:	**120 MINUTES**

Assignment 7: C# Basic Project Practice

Objectives:

>   »   Understand and practice with Classes, Object, Access Modifier, Constructors, this keyword.
>   »   Practice code in Visual Studio.
>   »   Follow coding convention.

Business needs:

>   »   TBD

Prerequisites:

>   »   Working environment: Visual Studio 2013 or higher.
>   »   Practice code in Visual Studio
>   »   Each exercise is one project inside 1 solution.

Technologies:

The product implements one or more technology:
>   »   C# Basic
>   »   Control of Flows
>   »   OOP

Technical Requirements:

>   »   Solution name must be NPL.M.A007.
>   »   Must create projects corresponding to each exercise:

   NPL.M.A007.Exercise1.

   NPL.M.A007.Exercise2.

1. **Exercise 1:**

Create a class called **Book** to represent a book. A Book should include four pieces of information as instance variables:
-   a book name.
-   an ISBN number.
-   an author name
-   a publisher name.

Your class should have a constructor that initializes the four instance variables.

In addition, provide a method named **GetBookInformation** that returns the description of the book as a String (the description should include all the information about the book). You should use *this* keyword in member methods and constructor.

Information display:

ISBN Number	Book Name	Author Name	Publisher Name
123456789	Harry Potter	J.K.Rowling	Kim Dong

Complete your code inside **NPL.M.A007.Exercise1.**
Run and check the result

Estimated time: 60 mins

2. **Exercise 2**

Create a class called **Car** that includes three pieces of information as instance variables
-   Speed (decimal);
-   RegularPrice (double);
-   Color (string);

This class also has a method GetSalePrice() (return double).

Create a sub-class of Car and name it as **Truck**. The Truck class has the following fields and methods:
-   Weight (int);
-   GetSalePrice() (return double); //If weight>2000,10% discount. Otherwise, 20% discount

Create a sub-class of Car class and name it as **Ford**. The Ford class has the following fields and methods.
-   Year (int);
-   ManufacturerDiscount (int);
-   GetSalePrice() (return double); //From the sale price computed from Car class, subtract the manufacturer Discount Create a sub-class of Car class and name it as Sedan. The Sedan class has the following fields and methods.
-   Length (int);
-   GetSalePrice() (double); // If length > 20 feet, 5% discount. Otherwise, 10% discount.

Create MyOwnAutoShop class which contains the main() method. Perform the following within the main() method.
-   Create an instance of Sedan class and initialize all the fields with appropriate values. Use base(...) method in the constructor for initializing the fields of the super class;
-   Create two instances of the Ford class and initialize all the fields with appropriate values. Use base(...) method in the constructor for initializing the fields of the super class;
-   Create two instances of Truck class and initialize all the fields with appropriate values. Use base(...) method in the constructor for initializing the fields of the super class.

Display the sale prices of all instance.

Complete your code inside **NPL.M.A007.Exercise2.**
Run and check the result

Estimated time: 60 mins