**ITSE 1330 – Intro to C#**

**Module 5 - Project 5**

**There are 3 projects for this Module. Place both project folders in a folder called FirstnameLastnameCSMod5. When you are done with all 3 projects, zip the folder and submit via Moodle. DO NOT ZIP EACH INDIVIDUAL PROJECT, only the outer folder.**

*Create a Windows Forms application named* ***FirstNameLastName5A***

*Save the application in the FirstnameLastnameCSMod5 folder.*

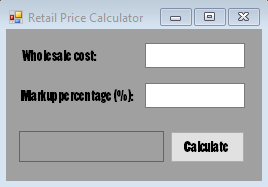
Create an application that lets the user enter an item’s wholesale cost and its markup percentage. It should then display the item’s retail price. For example:

If an item’s wholesale cost is $5.00 and its markup percentage is 100 percent, then the item’s retail price is $10.00.

If an item’s wholesale cost is $5.00 and its markup percentage is 50 percent, then the item’s retail price is $7.50.

The program should have a method named CalculateRetail that receives the wholesale cost and the markup percentage as arguments and returns the retail price of the item.

Example Form:



*Create a Windows Forms application named* ***FirstNameLastName5B***

*Save the application in the FirstnameLastnameCSMod5 folder.*

Joe’s Automotive performs the following routine maintenance services:

Oil change—$26.00

Lube job—$18.00

Radiator flush—$30.00

Transmission flush—$80.00

Inspection—$15.00

Muffler replacement—$100.00

Tire rotation—$20.00

Joe also performs other nonroutine services and charges for parts and labor ($20 per hour). Create an application that displays the total for a customer’s visit to Joe’s. The form should resemble the one shown below.

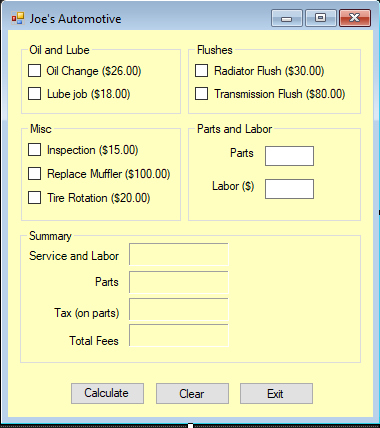
The application should have the following value-returning methods:

* OilLubeCharges—Returns the total charges for an oil change and/or a lube job, if any.
* FlushCharges—Returns the total charges for a radiator flush and/or a transmission flush, if any.
* MiscCharges—Returns the total charges for an inspection, muffler replacement, and/or a tire rotation, if any.
* OtherCharges—Returns the total charges for other services (parts and labor), if any.
* TaxCharges—Returns the amount of sales tax, if any. Sales tax is 6% and is charged only on parts. If the customer purchases services only, no sales tax is charged.
* TotalCharges—Returns the total charges.

The application should have the following void methods, called when the user clicks the Clear button:

* ClearOilLube—Clears the check boxes for oil change and lube job.
* ClearFlushes—Clears the check boxes for radiator flush and transmission flush.
* ClearMisc—Clears the check boxes for inspection, muffler replacement, and tire rotation.
* ClearOther—Clears the text boxes for parts and labor.
* ClearFees—Clears the labels that display the labels in the section marked Summary.

Example Form:



*Create a Windows Forms application named* ***FirstNameLastName5C***

*Save the application in the FirstnameLastnameCSMod5 folder.*

Create an application that lets the user play the game of Rock, Paper, Scissors against the computer. The program should work as follows:

When the program begins, a \*\***random number** in the range of 1 through 3 is generated. If the number is 1, then the computer has chosen rock. If the number is 2, then the computer has chosen paper. If the number is 3, then the computer has chosen scissors. (Do not display the computer’s choice yet.)

The user selects his or her choice of rock, paper, or scissors. To get this input you can use Button controls, or clickable PictureBox controls displaying some of the artwork that you will find in the Module 5 resources in moodle.

The computer’s choice is displayed.

A winner is selected according to the following rules:

* If one player chooses rock and the other player chooses scissors, then rock wins. (Rock smashes scissors.)
* If one player chooses scissors and the other player chooses paper, then scissors wins. (Scissors cuts paper.)
* If one player chooses paper and the other player chooses rock, then paper wins. (Paper wraps rock.)
* If both players make the same choice, the game must be played again to determine the winner.

Be sure to modularize the program into methods that perform each major task.

|  |
| --- |
| \*\***Random number code** |
| Class level |
| Random rand =- new Random(); |
| In method |
| rand.Next(3) – this will give you a random from 0, 1 and 2 |
| rand.Next(3) + 1 – this will give you a random number from 1, 2 and 3 |
| You can store this number in another variable or use it like this in a switch |

Example Form:

