Good job

CISP 41

Programming in C#

Project Evaluation Sheet

Student Name: Johnny Cho_____ Project Number: 3_____

Project Name: CISP 41's Auto Repair Shop_ Visual Studio Version: 2010_____

Date Due: 11/1/2012_____ Date Turned In: 11/1/2012_____

Above to be completed by student

Points (40 Possible)

Correctness/Efficiency:

Output is accurate — problems with sales tax in summary form — -2

Meets all requirements _____

Provide appropriate user interface _____

Logic is efficient _____

Documentation/Coding Style:

Project can be open from the submitted zip file _____

Folder is present and contains all necessary project files (no extra files) _____

Use required coding template _____

Use proper naming and spacing _____

Submit all requested information — use pdf instead of doc _____

Test Cases:

List all required test cases _____

Provide output forms for important test cases _____

Other issues: _____

Extra Credit: Not printing properly +2

Timeliness: _____

Project Score: 40

Project specification

This is a form for an Auto Repair Shop. It computes the costs. It has a preset cost of parts and the user can input the amount of labor done.

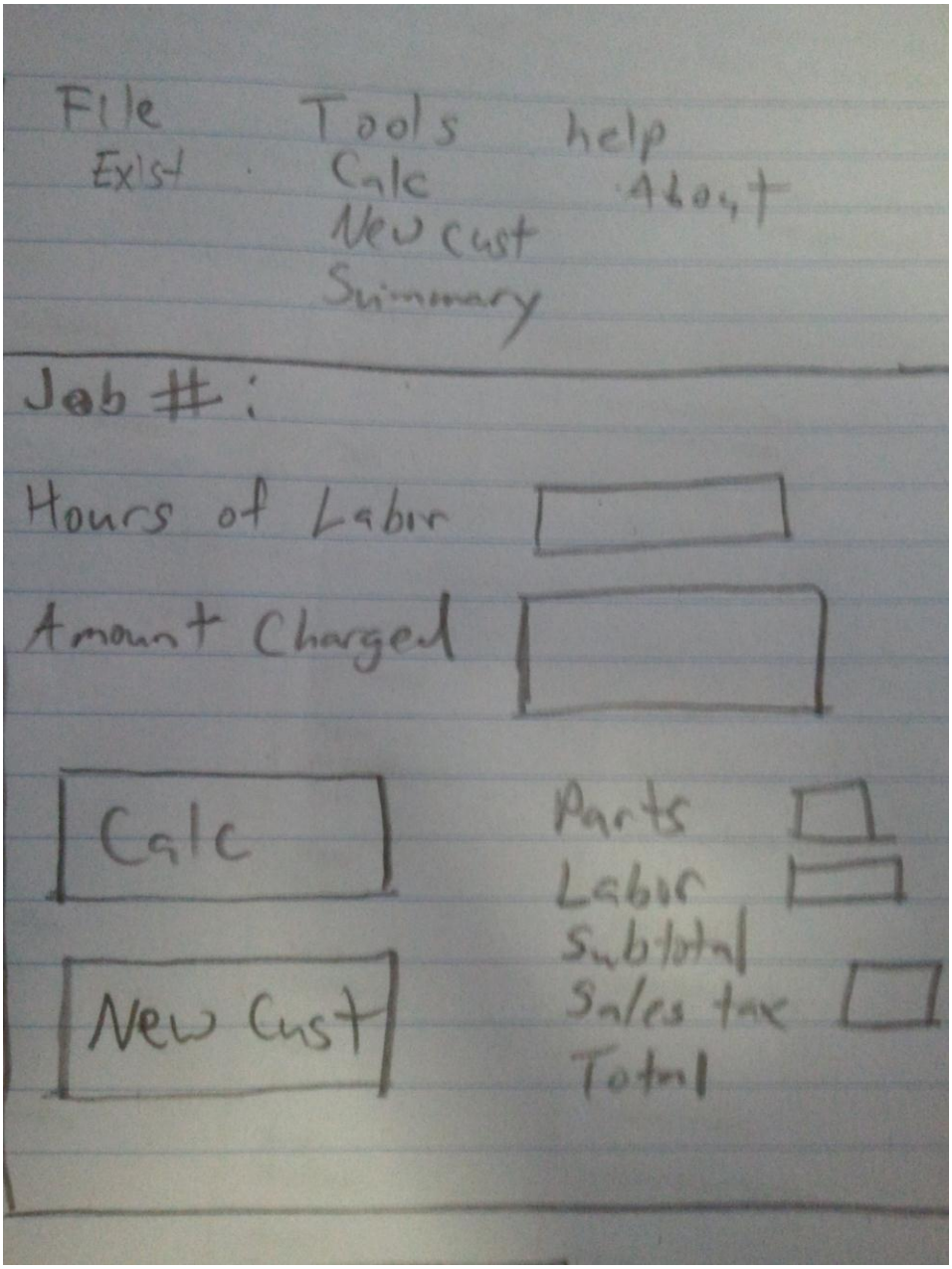
It has a summary form to display the total dollar amount of parts, labor, sales tax , and the total.

It also has a basic about form.

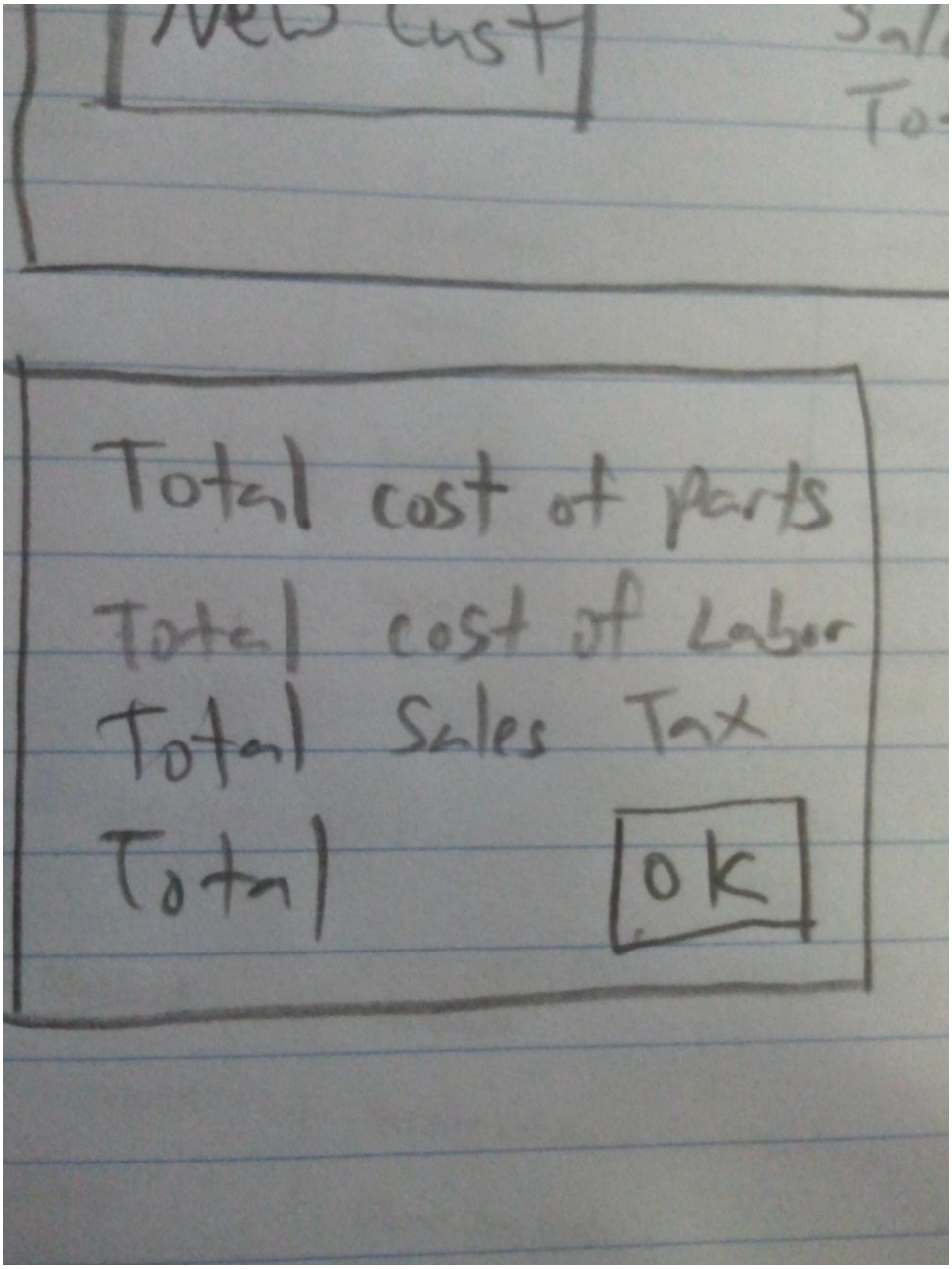
Project status

Works perfectly
I was somewhat unsure what "clears everything" means with the "New Customer" button so I commented out clearing the tax rate and sales tax rate.
Loops were the most difficult to program.
The printing format was not setup as well as I would have liked.

Sketch of user interface



Summary Form



CISP 41 Programming in C#

Objects and Properties Plan for _____Form

Object	Property	Setting
lblJobNum	**Name** **Text**	**lblJobNum** **Job Number:**
lblHoursLabor	**Name** **Text**	**lblHoursLabor** **Hours of Labor**
lblCharged	**Name** **Text**	**lblCharged Amount Charged for parts**
lblParts	**Name** **Text**	**lblParts** **Parts**
lblLabor	**Name** **Text**	**lblLabor** **Labor: $**
lblSubtotal	**Name** **Text**	**lblSubtotal** **Subtotal**
lblSalesTax	**Name** **Text**	**lblSalesTax** **Sales Tax**
lblTotal	**Name** **Text**	**lblTotal** **Total**
txtBoxHourLabor	**Name**	**txtBoxHourLabor**
listBox	**Name** **Items**	**listBox** **New Battery $100** **New Engine $1000** **Body Work $500** **Oil Change $40**
txtBoxParts	**Name**	**txtBoxParts**
txtBoxLabor	**Name**	**txtBoxLabor**
txtBoxSalesTax	**Name**	**txtBoxSalesTax**
btnCalculate	**Name** **Text**	**btnCalculate** **Calculate**
btnNewCust	**Name** **Text**	**btnNewCust** **New Customer**
menuStrip1	**Name**	**menuStrip1**
fileToolStripMenuItem	**Name** **Text**	**fileToolStripMenuItem** **File**
mnuExit	**Name** **Text**	**mnuExit** **Exit**
editToolStripMenuItem	**Name** **Text**	**editToolStripMenuItem** **Tools**
mnuCalculate	**Name** **Text**	**mnuCalculate** **Calculuate**
mnuClear	**Name** **Text**	**mnuClear** **New Customer**
mnuSummary	**Name** **Text**	**mnuSummary** **Summary**

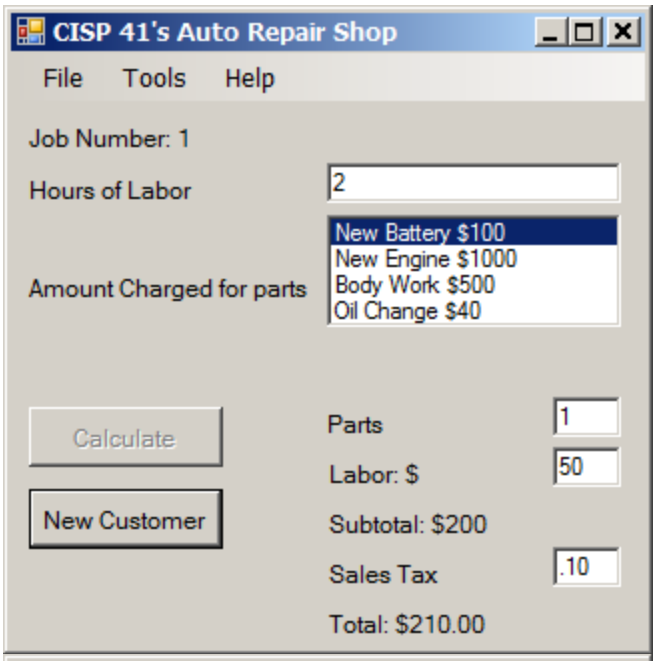
helpToolStripMenuItem	**Name** **Text**	**helpToolStripMenuItem** **Help**
mnuAbout	**Name** **Text**	**mnuAbout** **About**
summaryForm	**Name** **Text**	**summaryForm** **Summary**
lblParts	**Name** **Text**	**lblParts** **Total Cost of Parts:**
lblLabor	**Name** **Text**	**lblLabor** **Total Cost of Labor:**
lblSalesTax	**Name** **Text**	**lblSalesTax** **Total Sales Tax:**
lblTotal	**Name** **Text**	**lblTotal** **Total:**
btnSummaryOK	**Name** **Text**	**btnSummaryOK** **OK**
AboutBox1	**Name**	**AboutBox1**

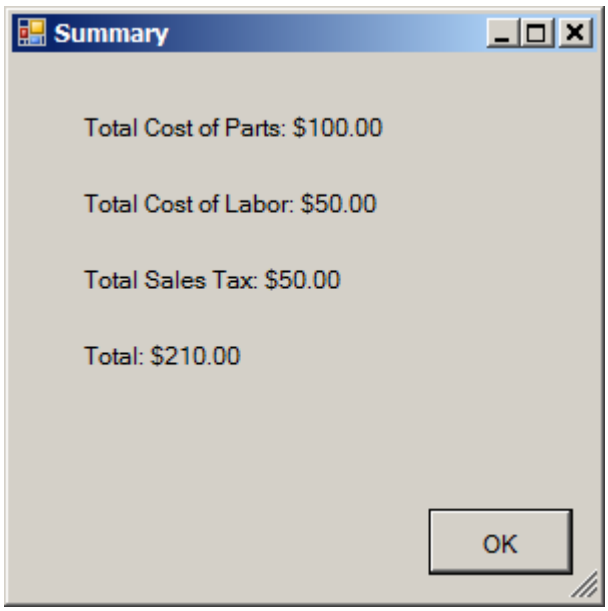
CISP 41 **Programming in C#**

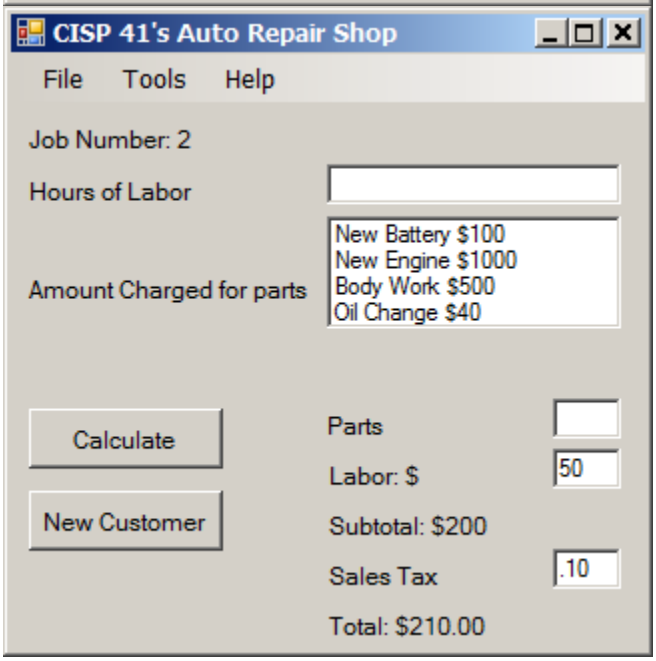
Event Plan for _____Form

Object	Event	Action - Pseudocode
mnuExit	Click	Exits program
getSumParts getSumLabor getSumSales getSumTotal	General method	Sends values of sums to summary form
calculate	General method	Calculates the costs
mnuCalculate	Click	Uses the calculate method
btnCalculate	Click	Uses the calculate method
clear	General method	Clears "everything"
mnuClear	Click	Uses clear method
btnNewCust	Click	Uses clear method
mnuAbout	Click	Brings up the about form
mnuSummary	Click	Brings up the summary form
putValues()	General method	Takes values and puts them into arrays
mnuPrint	Click	Prints document
printDocument1_PrintPage	General method	Sets up printing with loops
mnuPrintPreview	Click	Sees print preview
btnSummaryOK	Click	Closes summary form

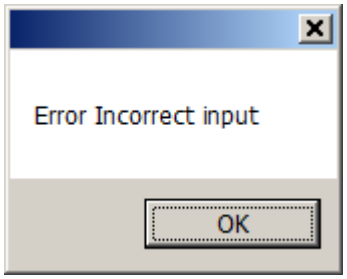
Test cases and captured screens

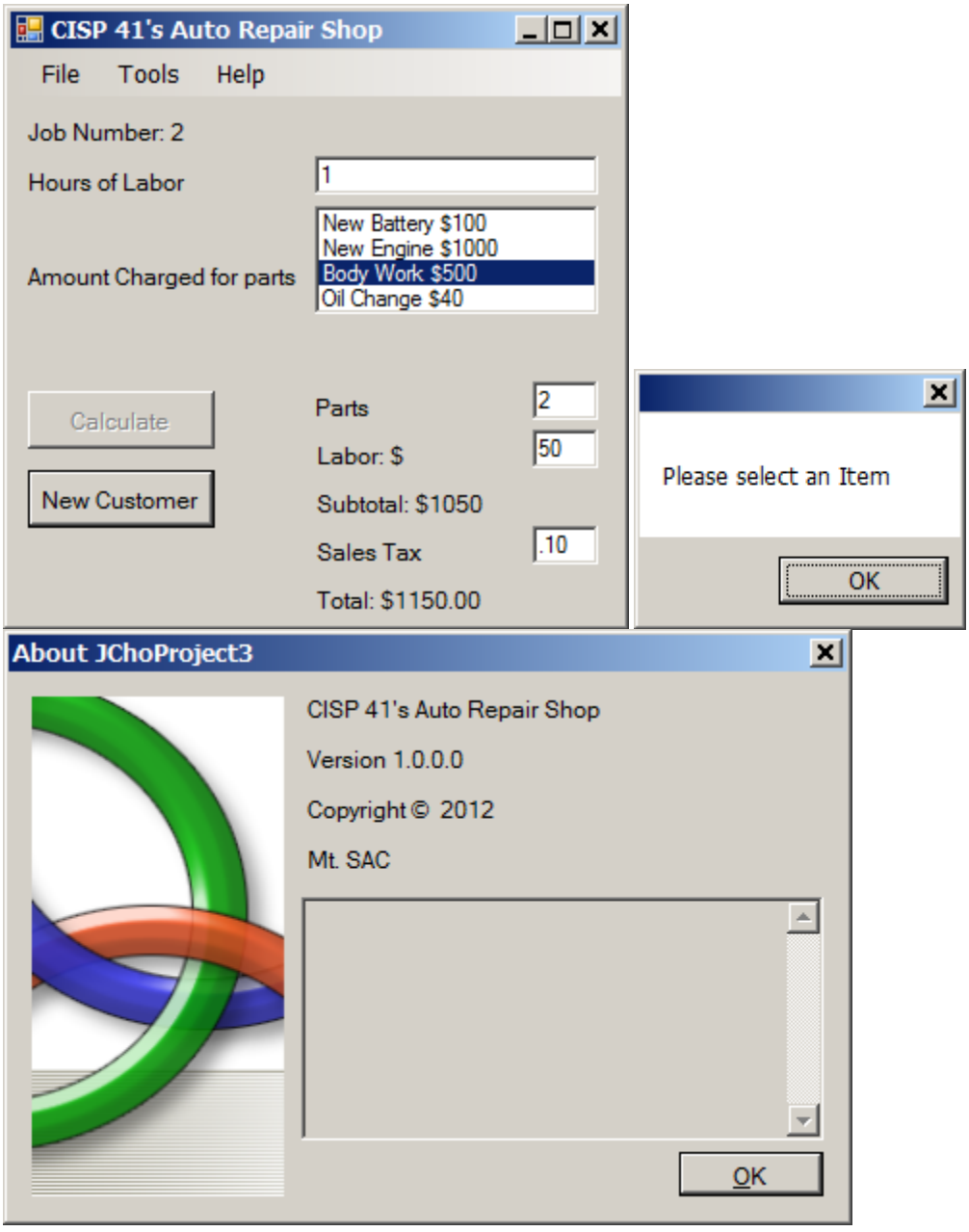






Display sales tax amount instead of tax rate.




CISP 41's Auto Repair Shop
File
Tools
Help
Job Number: 2
Hours of Labor
1
Amount Charged for parts
New Battery $100
New Engine $1000
Body Work $500
Oil Change $40
Calculate
New Customer
Parts
2
Labor: $
50
Subtotal: $1050
Sales Tax
.10
Total: $1150.00
Please select an Item
OK
About JChoProject3
CISP 41's Auto Repair Shop
Version 1.0.0.0
Copyright © 2012
Mt. SAC
OK

Source code

```
/*  Program:     Project 3
    Author:      Johnny Cho
    Class:      Cisp 41
    Date:       10/31/2012
    Description:

    I certify that the code below is my own work.

   Exception(s): N/A

*/

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace JChoExercise6._2
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            lblJobNum.Text = "Job Number: " + jobNumber.ToString();
        }

        private void mnuExit_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        int jobNumber = 1;
        decimal amountChargedParts;
        int hoursLabor;
        int numParts;
        decimal labor;
        decimal salesTax;
        decimal subTotal;
        decimal total;

        //decimal discountPercent;

        decimal[] costOfParts = new decimal[4]{100,1000,500,40};

        decimal[] arrayCParts = new decimal[10];
        decimal[] arrayCLabor = new decimal[10];
        decimal[] arrayCSales = new decimal[10];
        decimal[] arrayCTotal = new decimal[10];

        //variables to assign sums

        private static decimal sumParts;
        private static decimal sumLabor;
```

```
private static decimal sumSales;
private static decimal sumTotal;

public static decimal getSumParts
{
    get
    {
        return sumParts;
    }
}

public static decimal getSumLabor
{
    get
    {
        return sumLabor;
    }
}
public static decimal getSumSales
{
    get
    {
        return sumSales;
    }
}
public static decimal getSumTotal
{
    get
    {
        return sumTotal;
    }
}

private void calculate()
{
    try
    {
        hoursLabor = int.Parse(txtBoxHourLabor.Text);
        salesTax = decimal.Parse(txtBoxSalesTax.Text);
        numParts = int.Parse(txtBoxParts.Text);
        labor = decimal.Parse(txtBoxLabor.Text);

        if (listBox.SelectedIndex == -1)
        {
            MessageBox.Show("Please select an Item");
            return;
        }
        if (listBox.SelectedIndex == 0)
        {
            amountChargedParts = costOfParts[0];
        }
        if (listBox.SelectedIndex == 1)
        {
            amountChargedParts = costOfParts[1];
        }
        if (listBox.SelectedIndex == 2)
        {
            amountChargedParts = costOfParts[2];
        }
        if (listBox.SelectedIndex == 3)
        {
            amountChargedParts = costOfParts[3];
        }
        subTotal = (amountChargedParts * numParts) + (labor * hoursLabor);
```

```
            lblSubtotal.Text = "Subtotal: $" + subTotal;

            total = (amountChargedParts * numParts * salesTax) + subTotal;
            lblTotal.Text = "Total: $" + total;

            putValues();

            // Job Number Label
            jobNumber = jobNumber + 1;

            mnuCalculate.Enabled = false;
            btnCalculate.Enabled = false;
        }
        catch
        {
            MessageBox.Show("Error Incorrect input");
        }
    }

    private void mnuCalculate_Click(object sender, EventArgs e)
    {
        calculate();
    }
    private void btnCalculate_Click(object sender, EventArgs e)
    {
        calculate();
    }
    private void clear()
    {
        txtBoxHourLabor.Clear();
        txtBoxParts.Clear();
        //txtBoxLabor.Clear();
        //txtBoxSalesTax.Clear();
        txtBoxHourLabor.Focus();
        listBox.SelectedIndex = -1;

        // Job Number Label
        lblJobNum.Text = "Job Number: " + jobNumber.ToString();

        mnuCalculate.Enabled = true;
        btnCalculate.Enabled = true;
    }

    private void mnuClear_Click(object sender, EventArgs e)
    {
        clear();
    }
    private void btnNewCust_Click(object sender, EventArgs e)
    {
        clear();
    }
    private void mnuAbout_Click(object sender, EventArgs e)
    {
        AboutBox1 aboutForm = new AboutBox1();
        aboutForm.ShowDialog();
    }

    private void mnuSummary_Click(object sender, EventArgs e)
    {
        summaryForm aSummaryForm = new summaryForm();
        aSummaryForm.ShowDialog();
    }

    private void putValues()
```

```
        {
            //Places values into cost of parts array
            bool calcAParts = true;

            for (int i = jobNumber ; /*(i < arrayCParts.Length) &&*/ (calcAParts == true); )
            {
                arrayCParts[i] = amountChargedParts * numParts;
                calcAParts = false;

            }

            //Places values into labor cost array
            bool calcALabor = true;

            for (int j = jobNumber; (j < arrayCLabor.Length) && (calcALabor == true); j++)
            {
                arrayCLabor[j] = labor;
                calcALabor = false;
            }

            //Places values into total sales tax array
            bool calcASales = true;

            for (int k = jobNumber; (k < arrayCSales.Length) && (calcASales == true); k++)
            {
                arrayCSales[k] = (amountChargedParts * numParts * salesTax);
                calcASales = false;
            }

            //Places values into total cost array
            bool calcATotal = true;

            for (int l = jobNumber ; (l < arrayCTotal.Length) && (calcATotal == true); l++)
            {
                arrayCTotal[l] = total;
                calcATotal = false;
            }
            //Sum of the Arrays
            sumParts = arrayCParts.Sum();
            sumLabor = arrayCLabor.Sum();
            sumSales = arrayCSales.Sum();
            sumTotal = arrayCTotal.Sum();
        }

        private void mnuPrint_Click(object sender, EventArgs e)
        {
            printDocument1.Print();

        }

        private void printDocument1_PrintPage(object sender,
System.Drawing.Printing.PrintPageEventArgs e)
        {
            Font printFont = new Font("Arial", 12);
            float lineHeightFloat = printFont.GetHeight();
            float hPrintLocation;
            float vPrintLocation;
            string printLine;

            hPrintLocation = e.MarginBounds.Left;
            vPrintLocation = e.MarginBounds.Top;

            for (int indexInt = 1; indexInt < arrayCParts.Length; indexInt++)
            {
```

```
            printLine = arrayCParts[indexInt].ToString();

            e.Graphics.DrawString(printLine, printFont, Brushes.Black,
                hPrintLocation, vPrintLocation);

            vPrintLocation += lineHeightFloat;

        }
        printLine = "Total Parts: " + arrayCParts.Sum().ToString("C");
        e.Graphics.DrawString(printLine, printFont, Brushes.Black,
                hPrintLocation + 15, vPrintLocation);

        for (int indexInt = 1; indexInt < arrayCLabor.Length; indexInt++)
        {

            printLine = arrayCLabor[indexInt].ToString();


            e.Graphics.DrawString(printLine, printFont, Brushes.Black,
                hPrintLocation, vPrintLocation);


            vPrintLocation += lineHeightFloat;

        }
        printLine = "Total Labor: "+ arrayCLabor.Sum().ToString("C");
        e.Graphics.DrawString(printLine, printFont, Brushes.Black,
                hPrintLocation +15, vPrintLocation );

        for (int indexInt = 1; indexInt < arrayCSales.Length; indexInt++)
        {
            printLine = arrayCSales[indexInt].ToString();

            e.Graphics.DrawString(printLine, printFont, Brushes.Black,
                hPrintLocation, vPrintLocation);

            vPrintLocation += lineHeightFloat;

        }
        printLine ="Total Sales Tax: " + arrayCSales.Sum().ToString("C");
        e.Graphics.DrawString(printLine, printFont, Brushes.Black,
                hPrintLocation +15, vPrintLocation);

        for (int indexInt = 1; indexInt < arrayCTotal.Length; indexInt++)
        {
            printLine = arrayCTotal[indexInt].ToString();

            e.Graphics.DrawString(printLine, printFont, Brushes.Black,
                hPrintLocation, vPrintLocation);

            vPrintLocation += lineHeightFloat;

        }

        printLine = "Total: " + arrayCTotal.Sum().ToString("C");
        e.Graphics.DrawString(printLine, printFont, Brushes.Black,
                hPrintLocation + 15, vPrintLocation);
    }
    private void mnuPrintPreview_Click(object sender, EventArgs e)
    {
        printPreviewDialog1.Document = printDocument1;
        printPreviewDialog1.ShowDialog();
    }
}
```

Do not print the whole array; print only valid transactions.

```
}
```

Summary Form

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace JChoExercise6._2
{
    public partial class summaryForm : Form
    {
        public summaryForm()
        {
            InitializeComponent();
            lblParts.Text = "Total Cost of Parts: " + Form1.getSumParts.ToString("C");
            lblLabor.Text = "Total Cost of Labor: " + Form1.getSumLabor.ToString("C");
            lblSalesTax.Text = "Total Sales Tax: " + Form1.getSumLabor.ToString("C");
            lblTotal.Text = "Total: " + Form1.getSumTotal.ToString("C");
        }

        private void btnSummaryOK_Click(object sender, EventArgs e)
        {
            this.Close();
        }

    }
}
```

Incorrect assignment for total sales tax.