

## Lab 2 by Bryant Tunbutr

Shipping Form

ID (Six digits):

Number of Items

Weight:  lb.  oz.

Shipping Charges:

Total Sales:

Number of Transactions:

Shipping Form

ID (Six digits):

Number of Items

Weight:  lb.  oz.

Shipping Charges:

Total Sales:

Number of Transactions:

## Source code for ShippingForm.cs

```
/*
 * Project:          EX0304 - Exercise 3.4
 * Programmer:       Bryant Tunbutr
 * Date:            September 2012
 * Description:      Calculates and displays the charges for shipping a package.
 * I certify that the code below is my own work.
 */

/* Modified by T. Vo to remove Print function and add some class-level variables.
 * Modified by Bryant Tunbutr to add summary processing -- transactions and total.
 */

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

namespace EX0304
{
    public partial class ShippingForm : Form
    {
        const decimal SHIP_RATE_Decimal = 0.12m;
        private int transactionsInteger = 0;
        private decimal totalDecimal = 0.0m;

        public ShippingForm()
        {
            InitializeComponent();
        }

        private void calcButton_Click(object sender, EventArgs e)
        {
            // Calculate and display the shipping charges
            decimal shippingChargeDecimal;
            int poundsInteger, ouncesInteger;

            try
            {
                // Convert pounds if input was valid
                poundsInteger = int.Parse(poundsTextBox.Text);
                try
                {
                    // Convert ounces if input was valid
                    ouncesInteger = int.Parse(ouncesTextBox.Text);

                    // Add code to calculate shipping charges
                    shippingChargeDecimal = (SHIP_RATE_Decimal *
int.Parse(ouncesTextBox.Text) / 16
+ SHIP_RATE_Decimal * int.Parse(poundsTextBox.Text)) *
transactionsInteger;

                    // Add code to display shipping charges

                    chargesTextBox.Text = shippingChargeDecimal.ToString("C");

                    // Add code to update number transactions and total
                    transactionsInteger = int.Parse(numberOfItemsTextBox.Text);
                    totalDecimal = transactionsInteger + shippingChargeDecimal;
                }
            }
        }
    }
}
```

```

        // Add code to display number transactions and total

        totalSalesTextBox.Text = totalDecimal.ToString("C");
        transactionsTextBox.Text = transactionsInteger.ToString();
    }
    catch (FormatException)
    {
        // Handle if invalid amount entered for number of ounces
        MessageBox.Show("Please input the number of ounces (as a whole
number).", "Input Error", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        ouncesTextBox.Focus();
        ouncesTextBox.SelectAll();
    }
}
catch (FormatException)
{
    // Handle if invalid amount entered for number of pounds
    MessageBox.Show("Please input the number of pounds (as a whole number).",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
    poundsTextBox.Focus();
    poundsTextBox.SelectAll();
}
catch (Exception theException)
{
    // Handle all other exceptions.
    MessageBox.Show("Error: " + theException.Message);
}
}

private void clearButton_Click(object sender, EventArgs e)
{
    // Clear the form

    // Add code to clear all text boxes
    idTextBox.Clear();
    numberOfItemsTextBox.Clear();
    poundsTextBox.Clear();
    ouncesTextBox.Clear();
    chargesTextBox.Clear();
    totalSalesTextBox.Clear();
    transactionsTextBox.Clear();

    // Add code to set focus on idTextBox

    idTextBox.Focus();
}

private void exitButton_Click(object sender, EventArgs e)
{
    // End the program
    this.Close();
}
}
}

```

What would be the problem if we declared the two class-level variables (`transactionsInteger` and `totalDecimal`) in the `calculateButton_Click` method? You might want to try it out.

The problem is that the two variables are different, thus it adds extra steps because it is necessary to convert one of them in order to do a calculation. The program does not, unfortunately, allow implicit conversion.

You typically have to modify existing programs in the real world as part of your job as a developer. How is your experience with modifying this existing program?

It was honestly pretty challenging. The other labs had step by step instructions, but this one I had to think outside the box and do my best with the knowledge I have. For instance, I was not sure how much items cost, so I simply made them equal to the price of the quantity. It was a learning experience for sure.