PART ONE (INCLUDED CODE FILES)

namespace Retail\_Price\_Calculator

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

public void calculate()

{

try

{

if (IsValidData())

{

double originalPrice, discountPercentage, discountAmount, salePrice;

originalPrice = double.Parse(txtWholesale.Text);

discountPercentage = double.Parse(txtPercentage.Text);

discountPercentage = discountPercentage / 100;

discountAmount = originalPrice \* discountPercentage;

salePrice = originalPrice - discountAmount;

txtPrice.Text = salePrice.ToString("c");

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message + "\n\n" + ex.GetType().ToString() + "\n" + ex.StackTrace, "Exception");

}

}

private void button1\_Click(object sender, EventArgs e)

{

calculate();

}

public bool IsValidData()

{

return

IsPresent(txtWholesale, "Wholesale cost") &&

IsDecimal(txtWholesale, "Wholesale cost") &&

IsWithinRange(txtWholesale, "Wholesale cost", 0, 100000) &&

IsPresent(txtPercentage, "Markup Percentage") &&

IsDecimal(txtPercentage, "Markup Percentage") &&

IsWithinRange(txtPercentage, "Markup Percentage", 0, 100);

}

public bool IsPresent(TextBox textBox, string name)

{

if (textBox.Text == "")

{

MessageBox.Show(name + " is a required field.", "Entry Error");

textBox.Focus();

return false;

}

return true;

}

public bool IsDecimal(TextBox textBox, string name)

{

decimal number = 0m;

if (Decimal.TryParse(textBox.Text, out number))

{

return true;

}

else

{

MessageBox.Show(name + "must be a decimal value.", "Entry Error");

textBox.Focus();

return false;

}

}

public bool IsWithinRange(TextBox textBox, string name, decimal min, decimal max)

{

decimal number = Convert.ToDecimal(textBox.Text);

if (number < min || number > max)

{

MessageBox.Show(name + " must be between " + min + " and " + max + ".", "Entry Error");

textBox.Focus();

return false;

}

return true;

}

}

}

PART TWO (INCLUDED CODE FILES)

namespace Sum\_Evens

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

public void calculate()

{

double i, n, sum = 0;

n = double.Parse(txtInteger.Text);

for (i = 2; i <= n; i += 2)

{

sum += i;

}

txtResult.Text = sum.ToString("g");

}

private void button1\_Click(object sender, EventArgs e)

{

try

{

if (IsValidData())

{

calculate();

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message + "\n\n" + ex.GetType().ToString() + "\n" + ex.StackTrace, "Exception");

}

}

public bool IsValidData()

{

return

IsPresent(txtInteger, "Maximum Integer") &&

IsInt32(txtInteger, "Maximum Integer") &&

IsWithinRange(txtInteger, "Maximum Integer", 0, 1000);

}

public bool IsPresent(TextBox textBox, string name)

{

if (textBox.Text == "")

{

MessageBox.Show(name + " is a required field.", "Entry Error");

textBox.Focus();

return false;

}

return true;

}

public bool IsInt32(TextBox textBox, string name)

{

int number = 0;

if (Int32.TryParse(textBox.Text, out number))

{

return true;

}

else

{

MessageBox.Show(name + " must be an integer.", "Entry Error");

textBox.Focus();

return false;

}

}

public bool IsWithinRange(TextBox textBox, string name, decimal min, decimal max)

{

decimal number = Convert.ToDecimal(textBox.Text);

if (number < min || number > max)

{

MessageBox.Show(name + " must be between " + min + " and " + max + ".", "Entry Error");

textBox.Focus();

return false;

}

return true;

}

}

}