using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Present\_Value

{

public partial class PresentValueForm : Form

{

public PresentValueForm()

{

InitializeComponent();

}

private double getPresentValueCalc(double futureValue, int numYears, double interestRate)

{

//futurevalue over((1+interestRate)\*\*numYears)

double presentVal = (futureValue / (Math.Pow( (1 + interestRate) ,numYears)));

//send back the value to the orginal function to display

return presentVal;

}

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

{

//favorite method ever

this.Close();

}

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

{

// clear the form and focus back at the first field

futureValueTextBox.Text = "";

interestRateTextBox.Text = "";

yearsTextBox.Text = "";

presentValueTextBox.Text = "";

//because I'm fancy

presentValueLabel.Visible = false;

presentValueTextBox.Visible = false;

futureValueTextBox.Focus();

}

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

{

//declare variables to store the doubles of thes

int numYears;

double futureValue, interestRate, presentValue;

//error check the future value box

if(double.TryParse(futureValueTextBox.Text, out futureValue))

{

//error check the years value box

if (int.TryParse(yearsTextBox.Text,out numYears))

{

//error check the interest rate box

if (double.TryParse(interestRateTextBox.Text, out interestRate))

{

//convert the interest rate to a double since

interestRate = interestRate / 100;

//set the value that will be returned from the function

presentValue = getPresentValueCalc(futureValue, numYears, interestRate);

//convert that value to a string and show it to the user

presentValueTextBox.Text = presentValue.ToString("C");

//THEN show them the box since they can't tell the time difference anyway

presentValueLabel.Visible = true;

presentValueTextBox.Visible = true;

}

else

{

MessageBox.Show("You can only enter number values into the the interest rate field");

interestRateTextBox.Text = "";

interestRateTextBox.Focus();

}

}

else

{

MessageBox.Show("You can only enter whole numbers into the the years field");

yearsTextBox.Text = "";

yearsTextBox.Focus();

}

}

else

{

MessageBox.Show("You can only enter numbers into the the future value field");

futureValueTextBox.Text = "";

futureValueTextBox.Focus();

}

}

}

}