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 Calculater

{

public partial class Calculator : Form

{

string Result;

double EndResult;

string CheckForManipulation;

private decimal MemoryStore = 0;

public Calculator()

{

InitializeComponent();

}

private void Calculator\_Load(object sender, EventArgs e)

{

btn0.Click += new EventHandler(digitButton\_Click);

btn1.Click += new EventHandler(digitButton\_Click);

btn2.Click += new EventHandler(digitButton\_Click);

btn3.Click += new EventHandler(digitButton\_Click);

btn4.Click += new EventHandler(digitButton\_Click);

btn5.Click += new EventHandler(digitButton\_Click);

btn6.Click += new EventHandler(digitButton\_Click);

btn7.Click += new EventHandler(digitButton\_Click);

btn8.Click += new EventHandler(digitButton\_Click);

btn9.Click += new EventHandler(digitButton\_Click);

btnPlus.Click += new EventHandler(digitCalculate\_Click);

btnMinus.Click += new EventHandler(digitCalculate\_Click);

btnMult.Click += new EventHandler(digitCalculate\_Click);

btnDiv.Click += new EventHandler(digitCalculate\_Click);

btnMC.Click += new EventHandler(digitCalculate\_Click);

btnMR.Click += new EventHandler(digitCalculate\_Click);

btnMS.Click += new EventHandler(digitCalculate\_Click);

btnMPlus.Click += new EventHandler(digitCalculate\_Click);

}

void digitButton\_Click(object sender, EventArgs e)

{

Button ButtonThatWasPushed = (Button)sender;

txtResult.Text += ButtonThatWasPushed.Text;

}

void digitCalculate\_Click(object sender, EventArgs e)

{

Button ButtonThatWasPushed = (Button)sender;

string ButtonText = ButtonThatWasPushed.Text;

if (txtResult.Text != string.Empty)

{

double valueHolder1 = Convert.ToDouble(txtResult.Text);

}

if (ButtonText == "+")

{

Result = txtResult.Text;

CheckForManipulation = "Add";

txtResult.Clear();

txtResult.Focus();

}

else if (ButtonText == "-")

{

Result = txtResult.Text;

CheckForManipulation = "Substract";

txtResult.Clear();

txtResult.Focus();

}

else if (ButtonText == "\*")

{

Result = txtResult.Text;

CheckForManipulation = "Multiply";

txtResult.Clear();

txtResult.Focus();

}

else if (ButtonText == "/")

{

Result = txtResult.Text;

CheckForManipulation = "Division";

txtResult.Clear();

txtResult.Focus();

}

if (ButtonText == "MC")

{

//Memory Clear

this.MemoryStore = 0;

return;

}

if (ButtonText == "MR")

{

//Memory Recall

txtResult.Text = this.MemoryStore.ToString();

return;

}

if (ButtonText == "MS")

{

// Memory subtract

this.MemoryStore -= Convert.ToDecimal(this.EndResult);

return;

}

if (ButtonText == "M+")

{

// Memory add

//this.MemoryStore += decimal.Parse(this.Result);

this.MemoryStore += Convert.ToDecimal(this.EndResult);

return;

}

}

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

{

if (txtResult.Text != string.Empty && Result != string.Empty)

{

double valueHolder2 = Convert.ToDouble(txtResult.Text);

double chk = Convert.ToDouble(Result);

if (CheckForManipulation == "Add")

{

EndResult = chk + valueHolder2;

txtResult.Text = EndResult.ToString();

}

else if (CheckForManipulation == "Substract")

{

EndResult = chk - valueHolder2;

txtResult.Text = EndResult.ToString();

}

else if (CheckForManipulation == "Multiply")

{

EndResult = chk \* valueHolder2;

txtResult.Text = EndResult.ToString();

}

else if (CheckForManipulation == "Division")

{

if (valueHolder2 == 0)

{

txtResult.Text = "Cannot divide by Zero";

return;

}

EndResult = chk / valueHolder2;

txtResult.Text = EndResult.ToString();

}

}

}

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

{

if (txtResult.Text.Contains("."))

{

txtResult.Text = txtResult.Text;

}

else

{

txtResult.Text = txtResult.Text + ".";

}

}

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

{

//To find the reciprocal

if (txtResult.Text != string.Empty)

{

double chk = Convert.ToDouble("1") / Convert.ToDouble(txtResult.Text);

txtResult.Text = chk.ToString();

}

}

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

{

//To find the Inverse

if (txtResult.Text != string.Empty)

{

double chk = Convert.ToDouble(this.txtResult.Text);

chk = -chk;

txtResult.Text = chk.ToString();

}

}

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

{

//To find the Square root

if (txtResult.Text != string.Empty)

{

double SqrRoot = Math.Sqrt(Convert.ToDouble(this.txtResult.Text));

txtResult.Text = SqrRoot.ToString();

}

}

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

{

// to clear the Content

txtResult.Clear();

txtResult.Focus();

}

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

{

//backspace

if (txtResult.Text != string.Empty)

{

int txtlength=txtResult.Text.Length;

if (txtlength != 1)

{

txtResult.Text = txtResult.Text.Remove(txtlength - 1);

}

else

{

txtResult.Text = 0.ToString();

}

}

}

}

}