namespace DATAVIZ

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

//Joel Torney, 12/4, 202

// Click visualize data button to read a added text file

// The data from the text file is then displayed in an array

// The array is in bar graph form showing above avg., avg., and below avg.

// Each are color coded all in tab 1 (Visualization)

// the stats for the graph is displyed in the Stats tab

const int SZ = 1000;

int[] MyData = new int[SZ];

Graphics gr;

int j;

int Min, Max, sum=0;

Brush br;

double Avg;

//Create bar graph in Visualization tab

//shows data from file in stats tab

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

{

try

{//read file and convert to string

StreamReader rdr = new StreamReader("DATAVIZ.TXT");

for (j = 0; j < SZ; j++)

{

MyData[j] = int.Parse(rdr.ReadLine());

}

//call Stats and Draw method

Stats();

Draw();

}

catch (Exception Ex)

{

MessageBox.Show("Error!!" + Ex.Message);

}

}

//Calculate the Minimum, Maximum and Average

//Add File to Listbox and display Minimum, Maximum, and Average

private void Stats()

{

int sum = 0, Min = int.MaxValue, Max = int.MinValue;

LstOut2.Items.Clear();

//Calc Min,MAx,Avg from data file

int n;

for (n = 0; n < SZ; n++)

{

LstOut2.Items.Add( n.ToString() + ": " + MyData[n]);

sum += MyData[n];

if (MyData[n] < Min)

{

Min = MyData[n];

}

if (MyData[n] > Max)

{

Max = MyData[n];

}

//Converting the averrage number to a double

Avg = sum/(double)(SZ);

}

//Display Min, Max, and Avg

TxtOut.Text = "Avg: " + Avg + " Max: " + Max + " Min: " + Min;

}

//Creat graphics in visualization tab

private void Draw()

{

Brush br;

float x = 0,

y,

w = TabVisual.Width / (float)SZ,

h = 0;

gr = TabVisual.CreateGraphics();

//loop for data, create bar graph based on data from file

for (int i = 0; i < SZ; i++)

{

h = MyData[i];

y = TabVisual.Height - h;

x = x + w;

//deciding which bars are and arent above avg height

//Above average are green bars, below average are red bars

if (MyData[i] < Avg)

{

br = new SolidBrush(Color.Red);

}

else

{

br = new SolidBrush(Color.Green);

}

gr.FillRectangle(br, x, y, w, h);

}

//Creating black line at the average level in the data

br = new SolidBrush(Color.Black);

gr.FillRectangle(br, 0, TabVisual.Height - (float)Avg, TabVisual.Width, w);

}

}

}