ACITIVITY 1:

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 matrixAddition

{

public partial class firstMatrix : Form

{

public int[] values = new int[6];

public firstMatrix()

{

InitializeComponent();

}

public void storeValues()

{

values[0] = int.Parse(this.textBox1.Text);

values[1] = int.Parse(this.textBox2.Text);

values[2] = int.Parse(this.textBox3.Text);

values[3] = int.Parse(this.textBox4.Text);

values[4] = int.Parse(this.textBox5.Text);

values[5] = int.Parse(this.textBox6.Text);

}

private void contextMenuStrip1\_Opening(object sender, CancelEventArgs e)

{

}

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

{

storeValues();

// secondMatrix s = new secondMatrix();

secondMatrix sm = new secondMatrix();

sm.getValueOfMat1(values);

this.Hide();

sm.ShowDialog();

this.Dispose();

}

}

}

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 matrixAddition

{

public partial class secondMatrix : Form

{

public int[] prevValues = new int[6];

public int[] secondValue = new int[6];

public secondMatrix()

{

InitializeComponent();

}

public void getValueOfMat1(int[] arr)

{

prevValues = arr;

}

public void storeSecondValues()

{

secondValue[0] = int.Parse(this.textBox1.Text);

secondValue[1] = int.Parse(this.textBox2.Text);

secondValue[2] = int.Parse(this.textBox3.Text);

secondValue[3] = int.Parse(this.textBox4.Text);

secondValue[4] = int.Parse(this.textBox5.Text);

secondValue[5] = int.Parse(this.textBox6.Text);

}

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

{

storeSecondValues();

thirdMatrix tm = new thirdMatrix();

tm.getValueOfMat12(prevValues,secondValue);

tm.assignValues();

this.Hide();

tm.ShowDialog();

this.Dispose();

}

}

}

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 matrixAddition

{

public partial class thirdMatrix : Form

{

int[] firstsValues = new int[6];

int[] secondsValues = new int[6];

int[] thirdsValues = new int[6];

public thirdMatrix()

{

InitializeComponent();

}

public void getValueOfMat12(int[] arr,int[] arr2)

{

firstsValues = arr;

secondsValues = arr2;

}

public void assignValues()

{

for(int i=0; i<6; i++)

{

thirdsValues[i] = firstsValues[i] + secondsValues[i];

}

this.textBox1.Text = Convert.ToString(thirdsValues[0]);

this.textBox2.Text = Convert.ToString(thirdsValues[1]);

this.textBox3.Text = Convert.ToString(thirdsValues[2]);

this.textBox4.Text = Convert.ToString(thirdsValues[3]);

this.textBox5.Text = Convert.ToString(thirdsValues[4]);

this.textBox6.Text = Convert.ToString(thirdsValues[5]);

}

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

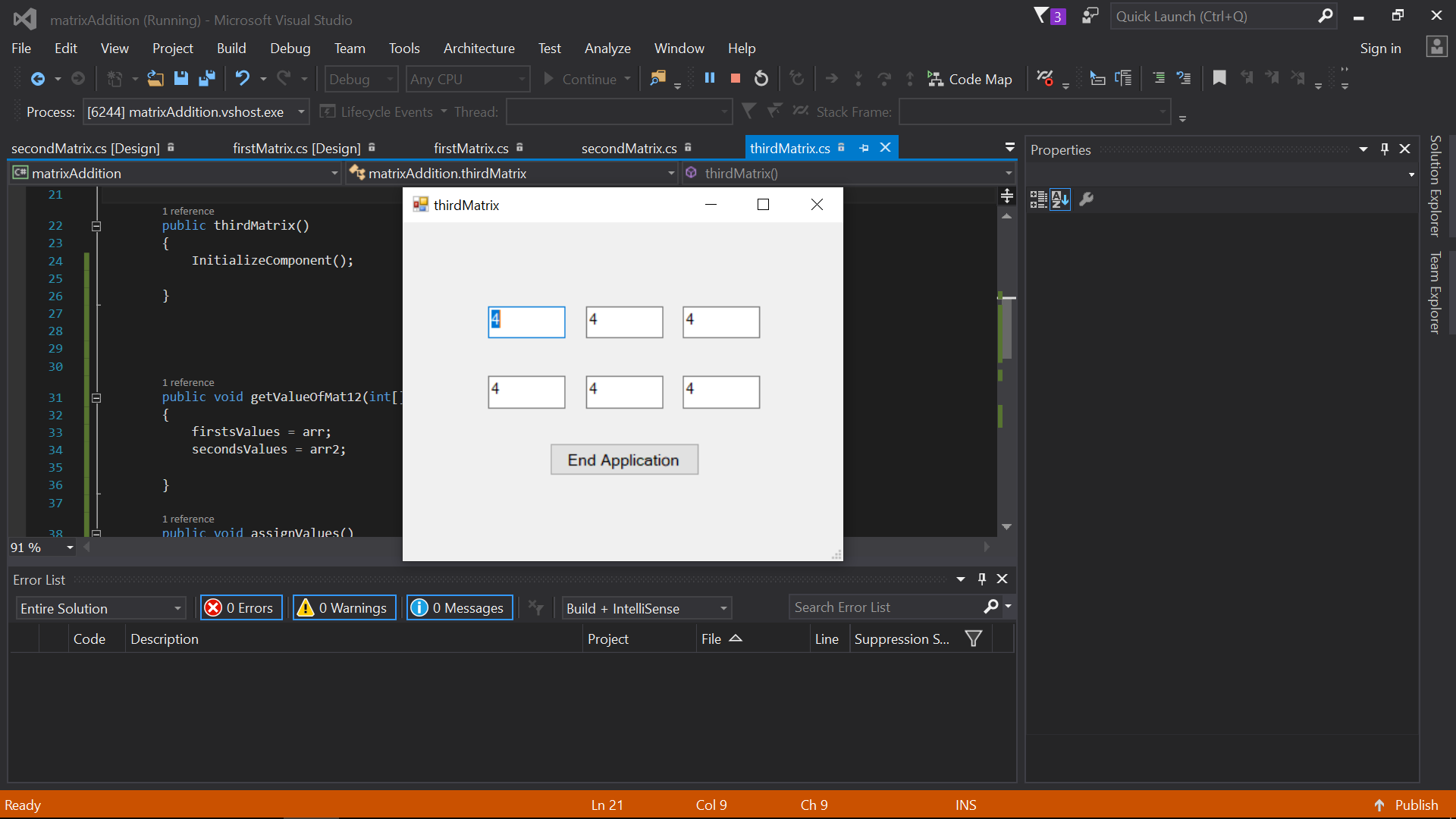
{

Application.Exit();

}

}

}



Activiy 2:

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 matrixMultiplication

{

public partial class frstMatrix : Form

{

public frstMatrix()

{

InitializeComponent();

}

public int[,] values = new int[3,3];

public void storeValues()

{

values[0,0] = int.Parse(this.textBox1.Text);

values[0,1] = int.Parse(this.textBox2.Text);

values[0,2] = int.Parse(this.textBox3.Text);

values[1,0] = int.Parse(this.textBox4.Text);

values[1,1] = int.Parse(this.textBox5.Text);

values[1,2] = int.Parse(this.textBox6.Text);

values[2, 0] = int.Parse(this.textBox7.Text);

values[2, 1] = int.Parse(this.textBox8.Text);

values[2, 2] = int.Parse(this.textBox9.Text);

}

private void contextMenuStrip1\_Opening(object sender, CancelEventArgs e)

{

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void button1\_Click\_1(object sender, EventArgs e)

{

storeValues();

// secondMatrix s = new secondMatrix();

scndMatrix sm = new scndMatrix();

sm.getValueOfMat1(values);

this.Hide();

sm.ShowDialog();

this.Dispose();

}

}

}

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 matrixMultiplication

{

public partial class scndMatrix : Form

{

int[,] prevValues = new int[3, 3];

int[,] secondValue = new int[3, 3];

public scndMatrix()

{

InitializeComponent();

}

public void getValueOfMat1(int[,] arr)

{

prevValues = arr;

}

public void storeSecondValues()

{

secondValue[0, 0] = int.Parse(this.textBox1.Text);

secondValue[0, 1] = int.Parse(this.textBox2.Text);

secondValue[0, 2] = int.Parse(this.textBox3.Text);

secondValue[1, 0] = int.Parse(this.textBox4.Text);

secondValue[1, 1] = int.Parse(this.textBox5.Text);

secondValue[1, 2] = int.Parse(this.textBox6.Text);

secondValue[2, 0] = int.Parse(this.textBox7.Text);

secondValue[2, 1] = int.Parse(this.textBox8.Text);

secondValue[2, 2] = int.Parse(this.textBox9.Text);

}

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

{

}

private void button1\_Click\_1(object sender, EventArgs e)

{

storeSecondValues();

thrdMatrix tm = new thrdMatrix();

tm.getValueOfMat12(prevValues, secondValue);

tm.assignValues();

this.Hide();

tm.ShowDialog();

this.Dispose();

}

}

}

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 matrixMultiplication

{

public partial class thrdMatrix : Form

{

public thrdMatrix()

{

InitializeComponent();

}

int[,] firstsValues = new int[3, 3];

int[,] secondsValues = new int[3, 3];

int[,] thirdsValues = new int[3, 3];

int value1, value2, value3;

int val1, val2, val3;

public void getValueOfMat12(int[,] arr, int[,] arr2)

{

firstsValues = arr;

secondsValues = arr2;

}

public void assignValues()

{

int i = 0, j = 0;

int m = 0, n = 0,l=0;

for (; i < 3; i++)

{

value1 = firstsValues[i, j];

value2 = firstsValues[i, j + 1];

value3 = firstsValues[i, j + 2];

for (; j < 3; j++)

{

val1 = secondsValues[l, j];

val2 = secondsValues[l + 1, j];

val3 = secondsValues[l + 2, j];

thirdsValues[m,n] = (value1 \* val1) + (value2 \* val2) + (value3 \* val3);

n++;

}

m++;

n = 0;

j = 0;

}

this.textBox1.Text = Convert.ToString(thirdsValues[0, 0]);

this.textBox2.Text = Convert.ToString(thirdsValues[0, 1]);

this.textBox3.Text = Convert.ToString(thirdsValues[0, 2]);

this.textBox4.Text = Convert.ToString(thirdsValues[1, 0]);

this.textBox5.Text = Convert.ToString(thirdsValues[1, 1]);

this.textBox6.Text = Convert.ToString(thirdsValues[1, 2]);

this.textBox7.Text = Convert.ToString(thirdsValues[2, 0]);

this.textBox8.Text = Convert.ToString(thirdsValues[2, 1]);

this.textBox9.Text = Convert.ToString(thirdsValues[2, 2]);

}

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

{

}

private void button1\_Click\_1(object sender, EventArgs e)

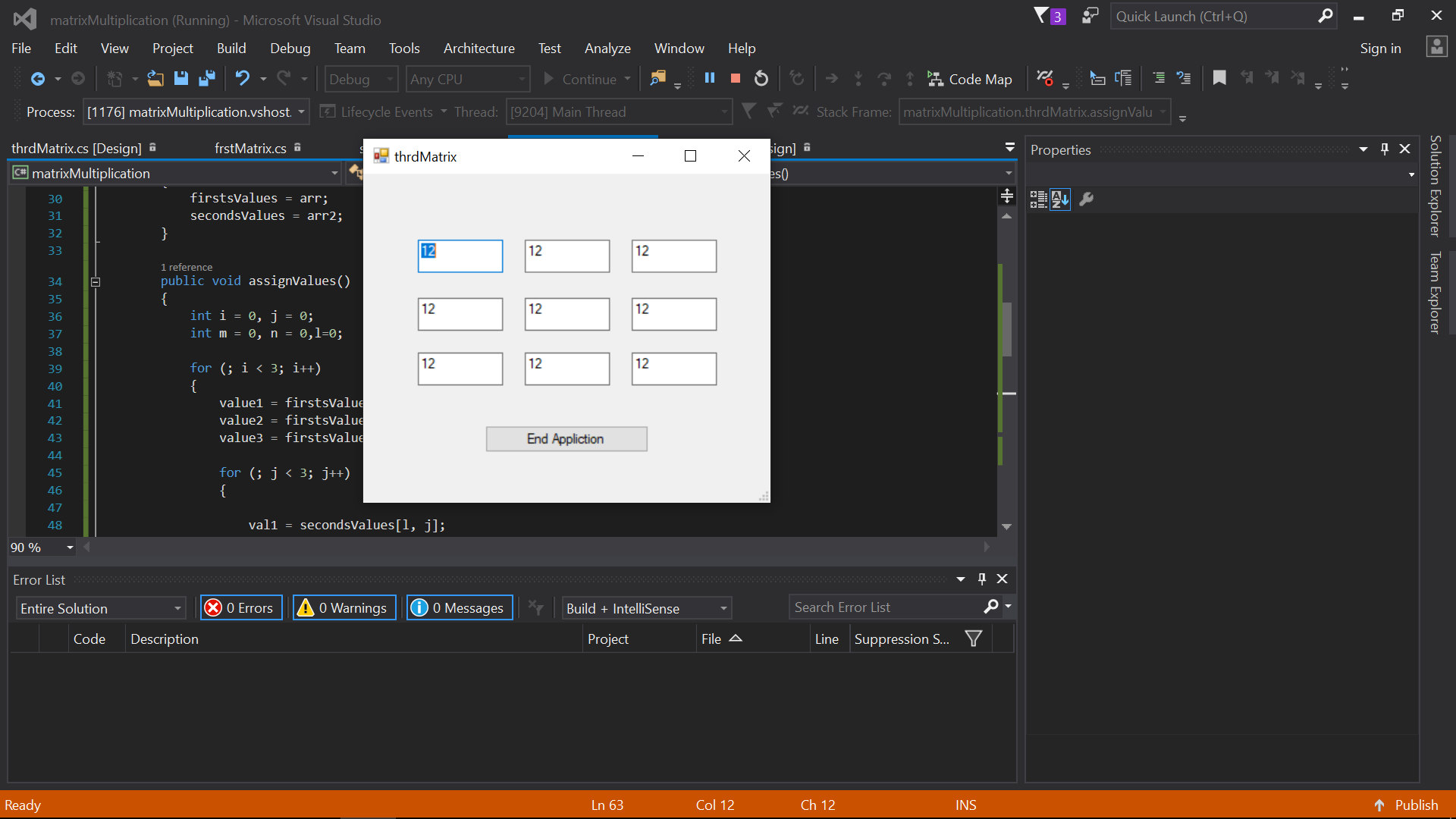
{

Application.Exit();

}

}

}



Activity 3

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 PictureViewer

{

public partial class frmImageViewer : Form

{

int counter=0;

string location = "c:\\users\\mushahid hussain\\documents\\visual studio 2015\\Projects\\PictureViewer\\PictureViewer\\picDirectory\\";

string[] pics = { "1.jpg","2.jpg","3.jpg","4.jpg","5.jpg" };

public frmImageViewer()

{

InitializeComponent();

picBox.ImageLocation = location+pics[counter];

counter = 1;

}

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

{

}

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

{

this.Dispose();

}

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

{

}

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

{

if (counter >= pics.Length)

{

label1.Text = "No more Pics";

label1.Visible = true;

btnNext.Enabled = false;

counter = pics.Length - 1;

}

else

{

label1.Visible = false;

btnPrev.Enabled = true;

picBox.ImageLocation = location + pics[counter++];

}

}

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

{

if (counter <= 0)

{

label1.Text = "No more Pics";

btnPrev.Enabled = false;

label1.Visible = true;

counter = 1;

}

else

{

btnNext.Enabled = true;

label1.Visible = false;

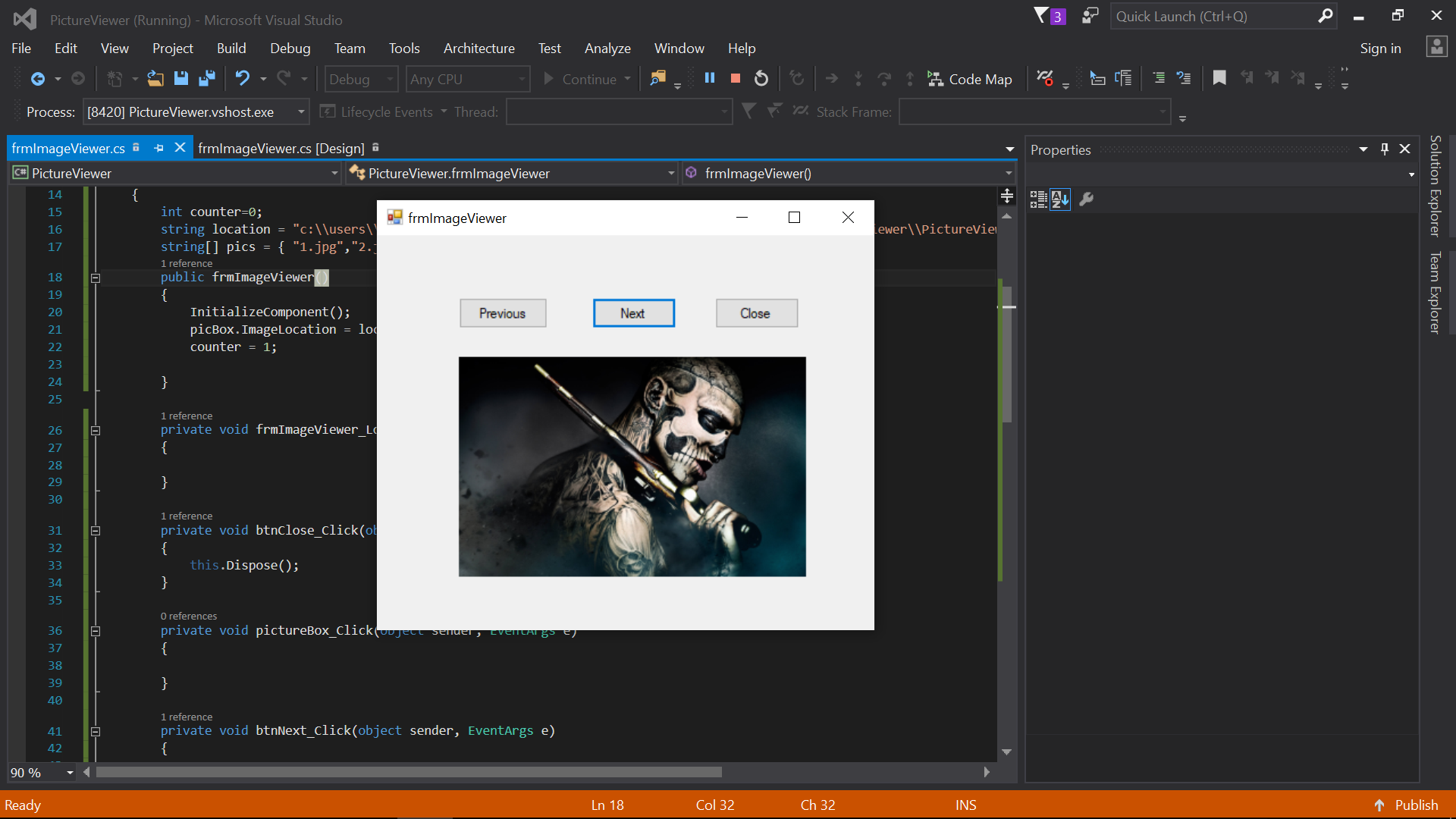
picBox.ImageLocation = location + pics[--counter];

}

}

}

}



Activity 4

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 puzzleGame

{

public partial class puzzleGame : Form

{

string location = "C:\\Users\\Mushahid Hussain\\Documents\\Visual Studio 2015\\Projects\\puzzleGame\\puzzleGame\\picsDir\\";

Random rand = new Random();

int val,index;

string[] arr = { "1.jpg", "2.jpg", "3.jpg", "4.jpg", "5.jpg", "6.jpg", "7.jpg", "8.jpg", "9.jpg", "10.jpg", "11.jpg", "12.jpg", "13.jpg" };

// List<Button> buttons;

List<int> number = new List<int>();

int win=0, lose=3;

public puzzleGame()

{

InitializeComponent();

val = rand.Next(0, 13);

this.btnPic.Image = Image.FromFile(location + arr[val]);

number.Add(val);

}

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

{

}

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

{

}

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

{

}

private async void btn1\_Click(object sender, EventArgs e)

{

// int index;

while (number.Contains(index))

index= rand.Next(0,13);

number.Add(index);

if (btn1.Image==null)

{

btn1.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn2\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn2.Image==null)

{

btn2.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn3\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if ( btn3.Image==null)

{

btn3.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn4\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn4.Image == null)

{

btn4.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn5\_Click(object sender, EventArgs e)

{

win++;

btn5.Image = Image.FromFile(location + arr[val]);

if (win > 2)

{

this.label1.Text = "Congrats, You won!";

await Task.Delay(1000);

Application.Exit();

}

}

private async void btn6\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn6.Image == null)

{

btn6.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn7\_Click(object sender, EventArgs e)

{

win++;

btn7.Image = Image.FromFile(location + arr[val]);

if (win > 2)

{

this.label1.Text = "Congrats, You won!";

await Task.Delay(1000);

Application.Exit();

}

}

private async void btn8\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn8.Image == null)

{

btn8.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn9\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn9.Image == null)

{

btn9.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn10\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn10.Image == null)

{

btn10.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn11\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn11.Image == null)

{

btn11.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn12\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (index != val && btn12.Image == null)

{

btn12.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn13\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn13.Image==null)

{

btn13.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn14\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn4.Image==null)

{

btn14.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

private async void btn15\_Click(object sender, EventArgs e)

{

win++;

btn15.Image = Image.FromFile(location + arr[val]);

if (win > 2)

{

this.label1.Text = "Congrats, You won!";

await Task.Delay(1000);

Application.Exit();

}

}

private async void btn16\_Click(object sender, EventArgs e)

{

while (number.Contains(index))

index = rand.Next(0, 13);

number.Add(index);

if (btn16.Image == null)

{

btn16.Image = Image.FromFile(location + arr[index]);

lose--;

label1.Text = "( Attemps Remaining " + lose + ")";

if (lose <= 0)

{

label1.Text = "( You Lose)";

await Task.Delay(1000);

Application.Exit();

}

}

}

}

}

