

*NIMRAH ALTAF ADAM*

*FESE-077*

*15th SEPTEMBER’2020*

#2

Q.1 Create an application in C# by using timer Control that generates racing between two cars on separate track bars, showing their speeds in label controls and final display the details of winning car.

FORM#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 WindowsFormsApp4

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

Form2 openForm = new Form2();

openForm.Show();

}

FORM#2

namespace WindowsFormsApp4

{

public partial class Form2 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form2\_Load(object sender, EventArgs e)

{

timer3.Enabled = true;

timer3.Start();

}

private void timer1\_Tick(object sender, EventArgs e)

{

Random rnd = new Random();

int rndBarcode = rnd.Next(700, 900);

timer1.Interval = rndBarcode;

textBox1.Text = rndBarcode.ToString();

pictureBox2.Location = new Point(pictureBox2.Location.X + 100, pictureBox2.Location.Y);

if (pictureBox2.Bounds.IntersectsWith(pictureBox5.Bounds))

{

timer1.Stop();

timer2.Stop();

MessageBox.Show("Hurray!! \nCar 2 Won This Race \nSpeed Of Car:" + textBox1.Text);

this.Close();

}

}

private void timer2\_Tick(object sender, EventArgs e)

{

pictureBox3.Location = new Point(pictureBox3.Location.X + 100, pictureBox3.Location.Y);

Random rand = new Random();

int randBarcode = rand.Next(700, 900);

timer2.Interval = randBarcode;

textBox2.Text = randBarcode.ToString();

if(pictureBox3.Bounds.IntersectsWith(pictureBox4.Bounds))

{

timer1.Stop();

timer2.Stop();

MessageBox.Show("Hurray!! \nCar 1 Won This Race \nSpeed Of Car:" + textBox2.Text);

this.Close();

}

}

int duration = 3;

private void timer3\_Tick(object sender, EventArgs e)

{

duration--;

textBox3.Text = duration.ToString();

if (duration == 0)

{

timer3.Stop();

textBox3.Visible = false;

timer1.Enabled = true;

timer1.Start();

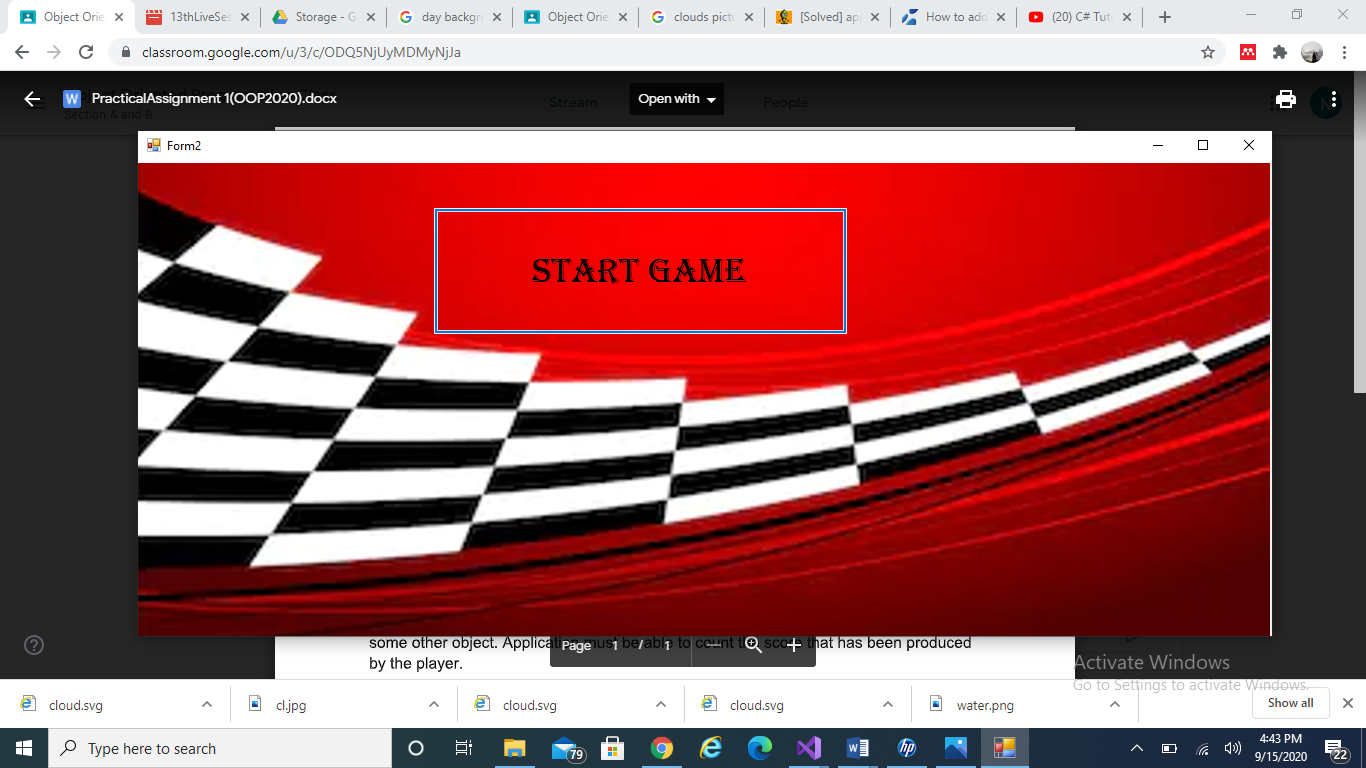
timer2.Enabled = true;

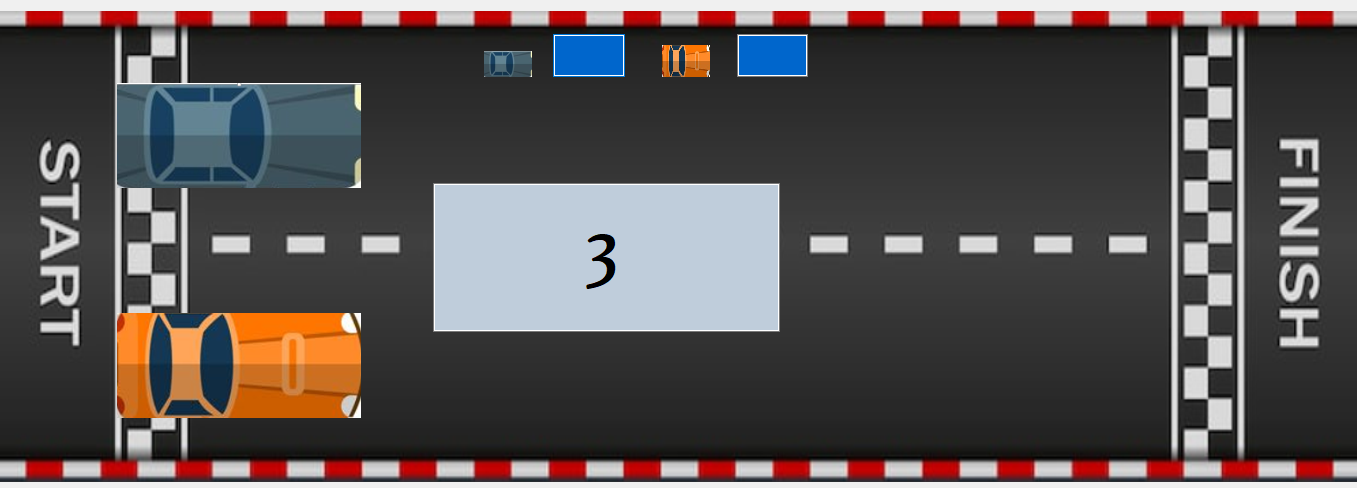
timer2.Start();

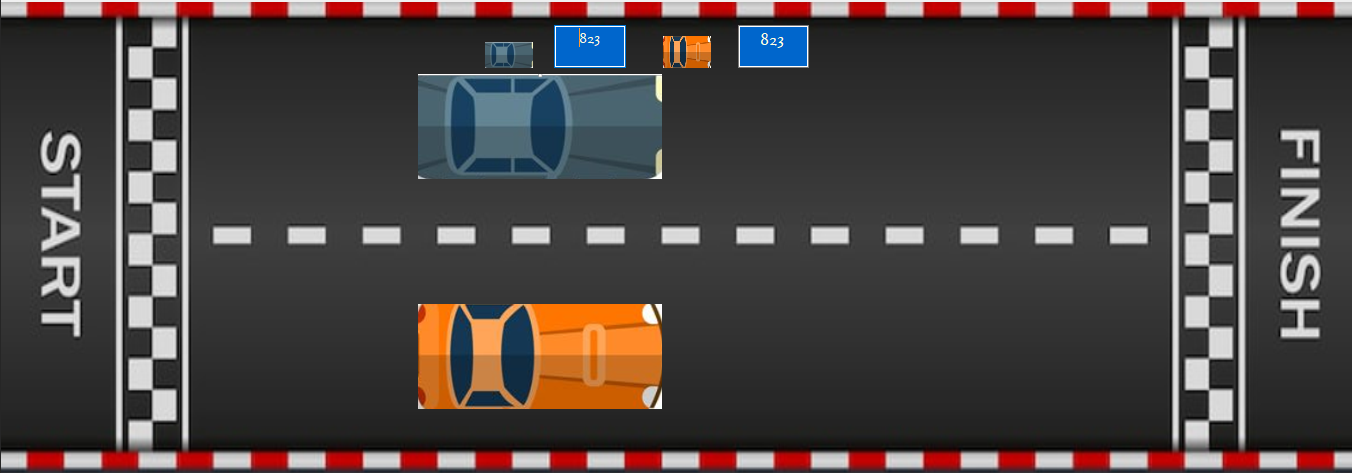
}

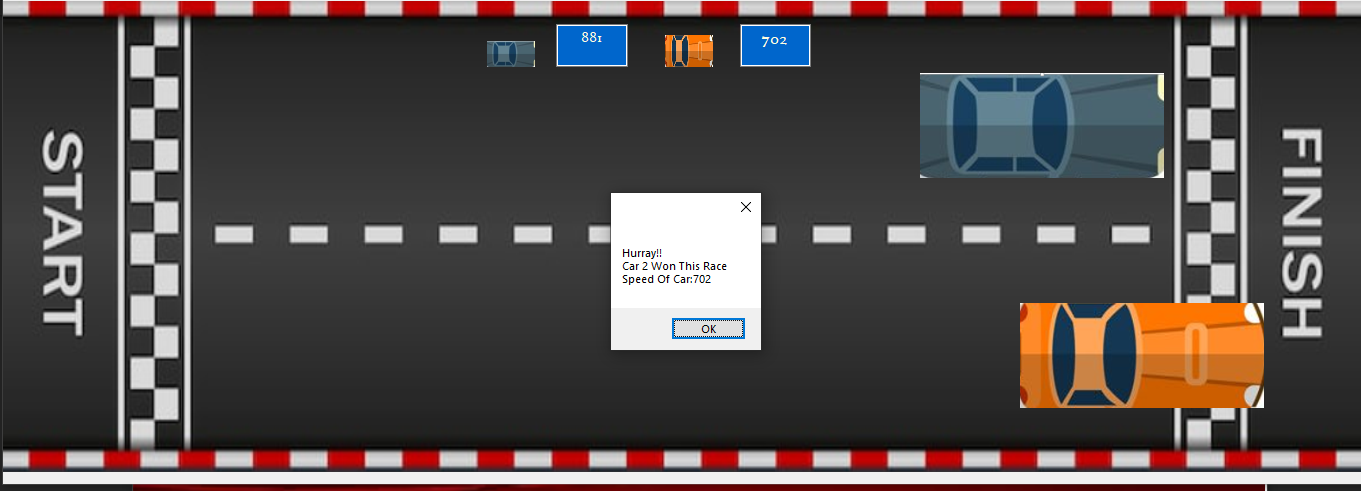
}

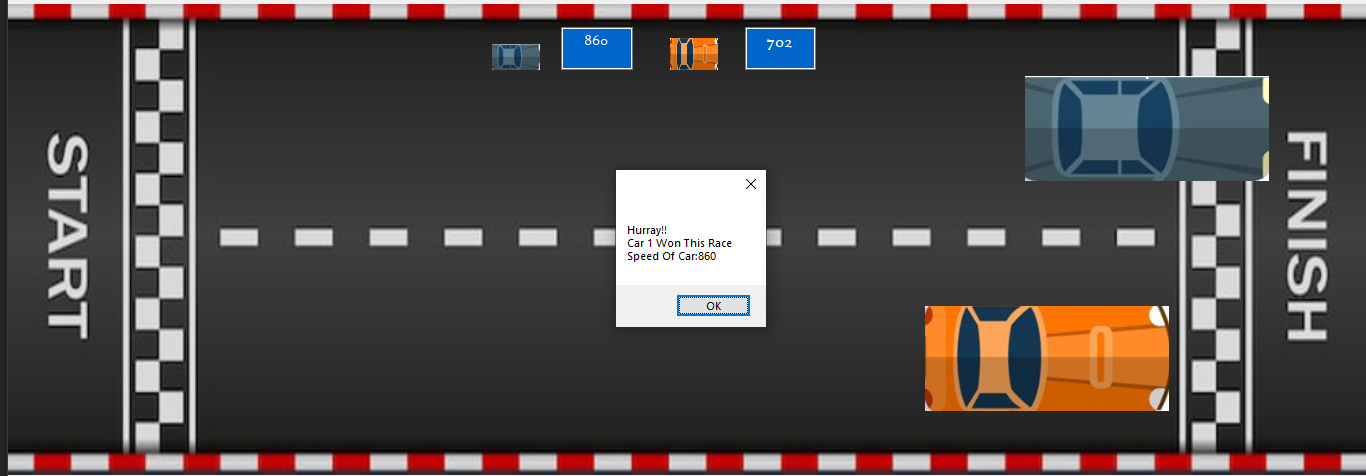
***OUTPUT***

******









Q.2 Create an application in C# that can create different controls (buttons, textboxes,

track bars and labels) dynamically at runtime that will generate an image of building by

using these controls.

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 WindowsFormsApp4

{

public partial class Form3 : Form

{

public Form3()

{

InitializeComponent();

CreateDynamicLabel(404, 254, 6, 333);

CreateDynamicLabel(510, 254, 6, 333);

CreateDynamicLabel(404, 587, 112, 6);

CreateDynamicLabel(404, 248, 15, 6);

CreateDynamicLabel(415, 233, 4, 15);

CreateDynamicLabel(415, 233, 15, 6);

CreateDynamicLabel(424, 119, 6, 120);

CreateDynamicLabel(424, 113, 15, 6);

CreateDynamicLabel(481, 113, 15, 6);//

CreateDynamicLabel(435, 88, 4, 25);

CreateDynamicLabel(435, 88, 50, 6);

CreateDynamicLabel(481, 88, 4, 25);

CreateDynamicLabel(490, 119, 6, 120);

CreateDynamicLabel(490, 233, 15, 6);

CreateDynamicLabel(501, 239, 4, 15);

CreateDynamicLabel(501, 248, 15, 6);

for (int i = 306; i <= 396; i += 30)

{

for (int j = 420; j <= 480; j += 30)

{

CreateDynamicButton(j, i);

}

}

for (int i = 128; i <= 188; i += 60)

{

for (int j = 434; j <= 476; j += 14)

{

CreateDynamicTextBox(j, i, 10, 53);

}

}

CreateDynamicTextBox(418, 269, 84, 10);

CreateDynamicTextBox(418, 283, 84, 10);

CreateDynamicTextBox(421, 423, 84, 10);

CreateDynamicTextBox(420, 437, 84, 10);

for (int i = 450; i <= 540; i += 30)

{

for (int j = 420; j <= 480; j += 30)

{

CreateDynamicButton(j, i);

}

}

}

private void CreateDynamicButton(int i, int j)

{

// Create a Button object

Button dynamicButton = new Button();

// Set Button properties

dynamicButton.Height = 24;

dynamicButton.Width = 24;

dynamicButton.BackColor = Color.White;

dynamicButton.ForeColor = Color.Black;

dynamicButton.Location = new Point(i, j);

Controls.Add(dynamicButton);

}

private void CreateDynamicLabel(int i, int j, int w, int h)

{

// Create a Label object

Label dynamicLabel = new Label();

// Set Label properties

dynamicLabel.Height = h;

dynamicLabel.Width = w;

dynamicLabel.BackColor = Color.Black;

dynamicLabel.Location = new Point(i, j);

Controls.Add(dynamicLabel);

}

private void CreateDynamicTextBox(int i, int j, int w, int h)

{

TextBox dynamicTextBox = new TextBox();

dynamicTextBox.Multiline = true;

dynamicTextBox.Height = h;

dynamicTextBox.Width = w;

dynamicTextBox.BackColor = Color.White;

dynamicTextBox.Location = new Point(i, j);

dynamicTextBox.BorderStyle = BorderStyle.Fixed3D;

Controls.Add(dynamicTextBox);

}

for (int i = 128; i <= 188; i += 60)

{

for (int j = 434; j <= 476; j += 14)

{

CreateDynamicTextBox(j, i, 10, 53);

}

}

CreateDynamicTextBox(418, 269, 84, 10);

CreateDynamicTextBox(418, 283, 84, 10);

CreateDynamicTextBox(421, 423, 84, 10);

CreateDynamicTextBox(420, 437, 84, 10);

for (int i = 450; i <= 540; i += 30)

{

for (int j = 420; j <= 480; j += 30)

{

CreateDynamicButton(j, i);

}

}

}

private void CreateDynamicButton(int i, int j)

{

// Create a Button object

Button dynamicButton = new Button();

// Set Button properties

dynamicButton.Height = 24;

dynamicButton.Width = 24;

dynamicButton.BackColor = Color.White;

dynamicButton.ForeColor = Color.Black;

dynamicButton.Location = new Point(i, j);

Controls.Add(dynamicButton);

}

private void CreateDynamicLabel(int i, int j, int w, int h)

{

// Create a Label object

Label dynamicLabel = new Label();

// Set Label properties

dynamicLabel.Height = h;

dynamicLabel.Width = w;

dynamicLabel.BackColor = Color.Black;

dynamicLabel.Location = new Point(i, j);

Controls.Add(dynamicLabel);

}

private void CreateDynamicTextBox(int i, int j, int w, int h)

{

TextBox dynamicTextBox = new TextBox();

dynamicTextBox.Multiline = true;

dynamicTextBox.Height = h;

dynamicTextBox.Width = w;

dynamicTextBox.BackColor = Color.White;

dynamicTextBox.Location = new Point(i, j);

dynamicTextBox.BorderStyle = BorderStyle.Fixed3D;

Controls.Add(dynamicTextBox);

}

private void CreateDynamicLabel(int i, int j, int w, int h)

{

// Create a Label object

Label dynamicLabel = new Label();

// Set Label properties

dynamicLabel.Height = h;

dynamicLabel.Width = w;

dynamicLabel.BackColor = Color.Black;

dynamicLabel.Location = new Point(i, j);

Controls.Add(dynamicLabel);

}

private void CreateDynamicTextBox(int i, int j, int w, int h)

{

TextBox dynamicTextBox = new TextBox();

dynamicTextBox.Multiline = true;

dynamicTextBox.Height = h;

dynamicTextBox.Width = w;

dynamicTextBox.BackColor = Color.White;

dynamicTextBox.Location = new Point(i, j);

dynamicTextBox.BorderStyle = BorderStyle.Fixed3D;

Controls.Add(dynamicTextBox);

}

***OUTPUT***



Q.3. Create an application in C sharp that will place different objects at random

locations over the form and they are moving with random speed and would be killed by

some other object. Application must be able to count the score that has been produced

by the player.

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 WindowsFormsApp4

{

public partial class Form5 : Form

{

bool moveLeft, moveRight, moveUp, moveDown;

int speed = 12;

public Form5()

{

InitializeComponent();

}

private void pictureBox1\_Click(object sender, EventArgs e)

{

}

int count = 0;

public Random r = new Random();

int cnt = 0;

private void timer2\_Tick(object sender, EventArgs e)

{

pictureBox2.Image = Image.FromFile(@"d:\7.jpg");

int x = r.Next(12, 699);

int y = r.Next(190, 247);

pictureBox2.Top = y;

pictureBox2.Left = x;

}

int duration = 3;

private void timer1\_Tick(object sender, EventArgs e)

{

duration--;

label2.Text = duration.ToString();

if (duration == 0)

{

timer1.Stop();

label2.Visible = false;

timer2.Enabled = true;

timer2.Start();

timer3.Enabled = true;

timer3.Start();

mainTimer.Enabled = true;

mainTimer.Start();

label3.Visible = true;

}

}

private void Form5\_Load(object sender, EventArgs e)

{

timer1.Enabled = true;

timer1.Start();

}

int dur = 30;

private void timer3\_Tick(object sender, EventArgs e)

{

dur--;

label5.Text = dur.ToString();

if (dur == 0)

{

timer1.Stop();

timer2.Stop();

timer3.Stop();

label3.Visible = false;

MessageBox.Show("GAME OVER!!", "Your score is " + label3.Text);

this.Close();

}

}

private void mainTimerEvent(object sender, EventArgs e)

{

if(moveLeft == true && player.Left > 0)

{

player.Left -=speed;

}

if (moveRight == true && player.Left < 730)

{

player.Left += speed;

}

if(moveUp == true && player.Top > 0)

{

player.Top -= speed;

}

if (moveDown == true && player.Top < 526)

{

player.Top += speed;

}

foreach(Control x in this.Controls)

{

if(x is PictureBox && (string)x.Tag == "Object")

{

if(player.Bounds.IntersectsWith(x.Bounds))

{

count++;

label3.Text = "Your Score" + count.ToString();

}

}

}

}

private void keyisdown(object sender, KeyEventArgs e)

{

if(e.KeyCode == Keys.Left)

{

moveLeft = true;

}

if (e.KeyCode == Keys.Right)

{

moveRight = true;

}

if (e.KeyCode == Keys.Down)

{

moveDown = true;

}

if (e.KeyCode == Keys.Up)

{

moveUp = true;

}

}

private void keyisup(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.Left)

{

moveLeft = false;

}

if (e.KeyCode == Keys.Right)

{

moveRight = false;

}

if (e.KeyCode == Keys.Down)

{

moveDown = false;

}

if (e.KeyCode == Keys.Up)

{

moveUp = false;

}

}

}

}

if(moveUp == true && player.Top > 0)

{

player.Top -= speed;

}

if (moveDown == true && player.Top < 526)

{

player.Top += speed;

}

foreach(Control x in this.Controls)

{

if(x is PictureBox && (string)x.Tag == "Object")

{

if(player.Bounds.IntersectsWith(x.Bounds))

{

count++;

label3.Text = "Your Score" + count.ToString();

}

}

}

}

private void keyisdown(object sender, KeyEventArgs e)

{

if(e.KeyCode == Keys.Left)

{

moveLeft = true;

}

if (e.KeyCode == Keys.Right)

{

moveRight = true;

}

if (e.KeyCode == Keys.Down)

{

moveDown = true;

}

if (e.KeyCode == Keys.Up)

{

moveUp = true;

}

}

private void keyisup(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.Left)

{

moveLeft = false;

}

if (e.KeyCode == Keys.Right)

{

moveRight = false;

}

if (e.KeyCode == Keys.Down)

{

moveDown = false;

}

if (e.KeyCode == Keys.Up)

{

moveUp = false;

}

}

}

}

private void keyisup(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.Left)

{

moveLeft = false;

}

if (e.KeyCode == Keys.Right)

{

moveRight = false;

}

if (e.KeyCode == Keys.Down)

{

moveDown = false;

}

if (e.KeyCode == Keys.Up)

{

moveUp = false;

}

}

}

}

**OUTPUT**

