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;

using System.Data.SqlClient;

using System.Diagnostics;

using System.Threading;

namespace VotingSystem

{

public partial class Topic : Form

{

public Topic()

{

InitializeComponent();

}

string strcon, strsql;

SqlConnection mycon;

SqlCommand command;

DataSet ds;

SqlDataAdapter da;

//Link database

string Key;//Key word

int TimeCount;

private bool DBConnect()

{

try

{

strcon = "Data Source=localhost;Initial Catalog=Voting;Integrated Security=True";

mycon = new SqlConnection(strcon);

mycon.Open();

MessageBox.Show("Link datebase is succesfully");

return true;

}

catch

{

MessageBox.Show("Link datebase is not succesfully");

return false;

}

//Check the database

}

private float X;// set a coordinate X

private float Y;// set a coordinate Y

private void setTag(Control cons)

{

foreach (Control con in cons.Controls)

{

con.Tag = con.Width + ":" + con.Height + ":" + con.Left + ":" + con.Top + ":" + con.Font.Size;

if (con.Controls.Count > 0)

setTag(con);

// Width + Height +Left + Top + Font

}

}

private void setControls(float newx, float newy, Control cons)

{

foreach (Control con in cons.Controls)

{

string[] mytag = con.Tag.ToString().Split(new char[] { ':' });

float a = Convert.ToSingle(mytag[0]) \* newx;

con.Width = (int)a;

a = Convert.ToSingle(mytag[1]) \* newy;

con.Height = (int)(a);

a = Convert.ToSingle(mytag[2]) \* newx;

con.Left = (int)(a);

a = Convert.ToSingle(mytag[3]) \* newy;

con.Top = (int)(a);

Single currentSize = Convert.ToSingle(mytag[4]) \* Math.Min(newx, newy);

con.Font = new Font(con.Font.Name, currentSize, con.Font.Style, con.Font.Unit);

if (con.Controls.Count > 0)

{

setControls(newx, newy, con);

}

}

// Store the original form and font size

}

void Form1\_Resize(object sender, EventArgs e)

{

float newx = (this.Width) / X;

float newy = this.Height / Y;

setControls(newx, newy, this);

this.Text = this.Width.ToString() + " " + this.Height.ToString();

// Reset the font when zooming in on the form

}

private void showDataGrid()

{

strsql = "select \* from Topic";

command = new SqlCommand(strsql, mycon);

command.ExecuteScalar();

ds = new DataSet();

da = new SqlDataAdapter(command);

da.Fill(ds, "topic");

Topic\_GD.DataSource = ds.Tables["topic"];

//Show information to form

}

private void Topic\_GD\_CellClick(object sender, DataGridViewCellEventArgs e)

{

Key = Topic\_GD.CurrentRow.Cells[1].Value.ToString();

TopicId\_txt.Text = Topic\_GD.CurrentRow.Cells[0].Value.ToString();

Topic\_txt.Text = Topic\_GD.CurrentRow.Cells[1].Value.ToString();

Candidate1.Text = Topic\_GD.CurrentRow.Cells[2].Value.ToString();

Candidate2.Text = Topic\_GD.CurrentRow.Cells[3].Value.ToString();

Candidate3.Text = Topic\_GD.CurrentRow.Cells[4].Value.ToString();

Candidate4.Text = Topic\_GD.CurrentRow.Cells[5].Value.ToString();

Dealine\_txt.Text = Topic\_GD.CurrentRow.Cells[6].Value.ToString();

Limited\_txt.Text = Topic\_GD.CurrentRow.Cells[7].Value.ToString();

Candidate1\_lab.Text = Topic\_GD.CurrentRow.Cells[8].Value.ToString();

Candidate2\_lab.Text = Topic\_GD.CurrentRow.Cells[9].Value.ToString();

Candidate3\_lab.Text = Topic\_GD.CurrentRow.Cells[10].Value.ToString();

Candidate4\_lab.Text = Topic\_GD.CurrentRow.Cells[11].Value.ToString();

// when users click data, it will transfor data from form to textbox

showCountDown(Topic\_GD.CurrentRow.Cells[12].Value.ToString(), Topic\_GD.CurrentRow.Cells[6].Value.ToString());

}

private void showCountDown(String previousTimeString, String deadline)

{

if (previousTimeString == null || previousTimeString.Length == 0)

{

return;

}

DateTime currentTime = DateTime.Now;

DateTime previousTime = Convert.ToDateTime(String.Copy(previousTimeString));

TimeSpan diff = currentTime.Subtract(previousTime);

Debug.WriteLine("Start Date : " + previousTime.ToString());

Debug.WriteLine("End Date ; " + currentTime.ToString());

Debug.WriteLine("Current Difference in minutes : " + diff.Minutes);

if (diff.Seconds >= Convert.ToInt32(Dealine\_txt.Text) \* 60)

{

Debug.WriteLine("Countdown Ends");

this.txthour.Text = (0).ToString();

this.txtmm.Text = (0).ToString();

this.txtss.Text = (0).ToString();

this.txtmss.Text = (0).ToString();

}

else

{

TimeCount = (Convert.ToInt32(Dealine\_txt.Text) \* 60000) - (diff.Seconds \* 6000);

showCountDownAnimation();

}

}

private void OK\_btn\_Click(object sender, EventArgs e)

{

this.txthour.Text = (Convert.ToInt32(txthour.Text)).ToString();

this.txtmm.Text = (Convert.ToInt32(txtmm.Text)).ToString();

this.txtss.Text = (Convert.ToInt32(txtss.Text)).ToString();

this.txtmss.Text = (Convert.ToInt32(txtmss.Text)).ToString();

if (this.txthour.Text == "0" & this.txtmm.Text == "0" & this.txtss.Text == "0" & this.txtmss.Text == "0")

{

this.timer1.Stop();

MessageBox.Show("Invalid vote");

}

if (Limited\_txt.Text[0].ToString() == "1") // when limited =1

{

if (Candidate1.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate1Voting = Candidate1Voting +1 where TopicId = '{1}'", Candidate1\_lab.Text,TopicId\_txt.Text, Key);// Voting candidate's vote +1

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate2.Checked == true) //

{

DBConnect();

strsql = string.Format("update Topic set Candidate2Voting = Candidate2Voting +1 where TopicId = '{1}'", Candidate2\_lab.Text,TopicId\_txt.Text, Key);// voting candidate2 vote +1

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate3.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate3Voting = Candidate3Voting +1 where TopicId = '{1}'", Candidate3\_lab.Text, TopicId\_txt.Text, Key);// voting candidate3 vote +1

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate4.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate4Voting = Candidate4Voting +1 where TopicId = '{1}'", Candidate4\_lab.Text, TopicId\_txt.Text, Key);// voting candidate4 vote +1

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

}

if (Limited\_txt.Text[0].ToString() == "2") // when limited is 2

{

if (Candidate1.Checked && Candidate2.Checked)

{

DBConnect();

strsql = string.Format("update Topic set Candidate1Voting = Candidate1Voting +1 , Candidate2Voting = Candidate2Voting +1 where TopicId = '{2}'", Candidate1\_lab.Text,Candidate2\_lab.Text, TopicId\_txt.Text, Key);// voting candidate1 and 2 vote +1

MessageBox.Show(strsql);

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate1.Checked && Candidate3.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate1Voting = Candidate1Voting +1 , Candidate3Voting = Candidate3Voting +1 where TopicId = '{2}'", Candidate1\_lab.Text, Candidate3\_lab.Text, TopicId\_txt.Text, Key);// voting candidate1 and 3 vote +1

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate1.Checked && Candidate4.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate1Voting = Candidate1Voting +1 , Candidate4Voting = Candidate4Voting +1 where TopicId = '{2}'", Candidate1\_lab.Text, Candidate4\_lab.Text, TopicId\_txt.Text, Key);// voting candidate1 and 4 vote +1

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate2.Checked && Candidate3.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate2Voting = Candidate2Voting +1 , Candidate3Voting = Candidate3Voting +1 where TopicId = '{2}'", Candidate2\_lab.Text, Candidate3\_lab.Text, TopicId\_txt.Text, Key);// voting candidate2 and 3 vote +1

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate2.Checked && Candidate4.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate2Voting = Candidate2Voting +1 , Candidate4Voting = Candidate4Voting +1 where TopicId = '{2}'", Candidate2\_lab.Text, Candidate4\_lab.Text, TopicId\_txt.Text, Key);// voting candidate2 and 4 vote +1

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate3.Checked && Candidate4.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate3Voting = Candidate3Voting +1 , Candidate4Voting = Candidate4Voting +1 where TopicId = '{2}'", Candidate3\_lab.Text, Candidate4\_lab.Text, TopicId\_txt.Text, Key);// voting candidate3 and 4 vote +1

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

}

if (Limited\_txt.Text[0].ToString() == "3") // when limited is 3

{

if (Candidate1.Checked && Candidate2.Checked && Candidate3.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate1Voting = Candidate1Voting +1 , Candidate2Voting = Candidate2Voting +1 , Candidate3Voting = Candidate3Voting +1 where TopicId = '{3}'", Candidate1\_lab.Text, Candidate2\_lab.Text, Candidate3\_lab.Text, TopicId\_txt.Text, Key);// voting candidate1 and 2 and 3vote +1

MessageBox.Show(strsql);

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate1.Checked && Candidate2.Checked && Candidate4.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate1Voting = Candidate1Voting +1 , Candidate2Voting = Candidate2Voting +1 , Candidate4Voting = Candidate4Voting +1 where TopicId = '{3}'", Candidate1\_lab.Text, Candidate2\_lab.Text, Candidate4\_lab.Text, TopicId\_txt.Text, Key);// voting candidate1 and 2 and 4vote +1

MessageBox.Show(strsql);

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate2.Checked && Candidate3.Checked && Candidate4.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate2Voting = Candidate2Voting +1 , Candidate3Voting = Candidate3Voting +1 , Candidate4Voting = Candidate4Voting +1 where TopicId = '{3}'", Candidate2\_lab.Text, Candidate3\_lab.Text, Candidate4\_lab.Text, TopicId\_txt.Text, Key);// voting candidate3 and 2 and 4vote +1

MessageBox.Show(strsql);

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

if (Candidate1.Checked && Candidate3.Checked && Candidate4.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate1Voting = Candidate1Voting +1 , Candidate3Voting = Candidate3Voting +1 , Candidate4Voting = Candidate4Voting +1 where TopicId = '{3}'", Candidate1\_lab.Text, Candidate3\_lab.Text, Candidate4\_lab.Text, TopicId\_txt.Text, Key);// voting candidate1 and 3 and 4 vote +1

MessageBox.Show(strsql);

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

}

if (Limited\_txt.Text[0].ToString() == "4") //when limited is 4

{

if (Candidate1.Checked && Candidate2.Checked && Candidate3.Checked && Candidate4.Checked == true)

{

DBConnect();

strsql = string.Format("update Topic set Candidate1Voting = Candidate1Voting +1 , Candidate2Voting = Candidate2Voting +1 , Candidate3Voting = Candidate3Voting +1 , Candidate4Voting = Candidate4Voting +1where TopicId = '{4}'", Candidate1\_lab.Text, Candidate2\_lab.Text, Candidate3\_lab.Text, Candidate4\_lab.Text, TopicId\_txt.Text, Key);// voting candidate1 and 2 and 3 and 4 vote +1

MessageBox.Show(strsql);

command = new SqlCommand(strsql, mycon);

try

{

command.ExecuteScalar();

MessageBox.Show("Successfully Voting.");

}

catch

{

MessageBox.Show("Voting Error.");

}

//Check register

finally

{

mycon.Close();

}

//Close the database

}

}

}

private void Topic\_GD\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

int Timecount = 0;

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

{

if (Timecount >= 0)

{

this.txthour.Text = Timecount .ToString();

this.txtmm.Text = Timecount.ToString();

this.txtss.Text = Timecount.ToString();

this.txtmss.Text = Timecount.ToString();

if (Timecount == 0)

{

txthour.ForeColor = Color.Red;

txtmm.ForeColor = Color.Red;

txtss.ForeColor = Color.Red;

txtmss.ForeColor = Color.Red;

}

Timecount -= 10;

}

}

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

{

DBConnect();

showDataGrid();

int count = ds.Tables["Topic"].Rows.Count;//get this row

lbl9.Text = count.ToString() + " records in the table.";

Topic\_GD.RowHeadersVisible = false;

this.Topic\_GD.Columns[0].Width = 80;

this.Topic\_GD.Columns[1].Width = 80;

this.Topic\_GD.Columns[2].Width = 80;

this.Topic\_GD.Columns[3].Width = 80;

this.Topic\_GD.Columns[4].Width = 80;

this.Topic\_GD.Columns[5].Width = 80;

this.Topic\_GD.Columns[6].Width = 80;

this.Topic\_GD.Columns[7].Width = 80;

this.Topic\_GD.Columns[8].Width = 80;

this.Topic\_GD.Columns[9].Width = 80;

this.Topic\_GD.Columns[10].Width = 80;

this.Topic\_GD.Columns[11].Width = 80;

// update information to database

Topic\_GD.ColumnHeadersDefaultCellStyle.Font = new Font("Tahoma", 7, FontStyle.Bold);

// Change typeface

mycon.Close();

}

private void showCountDownAnimation()

{

this.timer1.Stop();

this.timer1.Interval = 1;

this.timer1.Start();

Thread counter = new Thread(Counter);

counter.Start();

Control.CheckForIllegalCrossThreadCalls = false;//放弃捕获对错误线程的调用，否则在线程中无法调用控件名

this.Dealine\_txt.ReadOnly = true;

txthour.ForeColor = Color.Black;

txtmm.ForeColor = Color.Black;

txtss.ForeColor = Color.Black;

txtmss.ForeColor = Color.Black;

}

public void Counter()

{

try

{

while (TimeCount >= 0)

{

this.txthour.Text = (TimeCount / 3600000).ToString();

this.txtmm.Text = ((TimeCount / 60000) % 60).ToString();

this.txtss.Text = ((TimeCount / 1000) % 60).ToString();

this.txtmss.Text = (TimeCount % 1000).ToString();

if (TimeCount == 0)

{

txthour.ForeColor = Color.Red;

txtmm.ForeColor = Color.Red;

txtss.ForeColor = Color.Red;

txtmss.ForeColor = Color.Red;

this.Dealine\_txt.ReadOnly = false;

try

{

Thread currthread = Thread.CurrentThread;

currthread.Abort();// 终止当前进程，会触发ThreadAbortException异常，从而终止进程，所以下面需要捕获该异常才能终止进程

}

catch (ThreadAbortException) { }

}

TimeCount -= 1;

Thread.Sleep(1);

}

}

catch

{

MessageBox.Show("Finished");

}//处理异常关闭情况下的异常问题

}

private void Refresh\_btn\_Click(object sender, EventArgs e)

{

}

private void Home\_btn\_Click(object sender, EventArgs e)

{

Login login = new Login();

this.Hide();

login.ShowDialog(this);

//transfor "Login"

}

private void button2\_Click(object sender, EventArgs e)

{

Application.Exit();

//colse this form

}

private void txtmss\_TextChanged(object sender, EventArgs e)

{

}

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

{

txthour.Text = Time.i.ToString();

txtmm.Text = Time.a.ToString();

txtss.Text = Time.b.ToString();

txtmss.Text = Time.c.ToString();

DBConnect();

showDataGrid();

Topic\_GD.RowHeadersVisible = false;

this.Topic\_GD.Columns[0].Width = 80;

this.Topic\_GD.Columns[1].Width = 80;

this.Topic\_GD.Columns[2].Width = 80;

this.Topic\_GD.Columns[3].Width = 80;

this.Topic\_GD.Columns[4].Width = 80;

this.Topic\_GD.Columns[5].Width = 80;

this.Topic\_GD.Columns[6].Width = 80;

this.Topic\_GD.Columns[7].Width = 80;

this.Topic\_GD.Columns[8].Width = 80;

this.Topic\_GD.Columns[9].Width = 80;

this.Topic\_GD.Columns[10].Width = 80;

this.Topic\_GD.Columns[11].Width = 80;

Topic\_GD.ColumnHeadersDefaultCellStyle.Font = new Font("Tahoma", 7, FontStyle.Bold);

// change font type

this.Resize += new EventHandler(Form1\_Resize);

X = this.Width;

Y = this.Height;

setTag(this);

Form1\_Resize(new object(), new EventArgs());

}

}

}