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;

namespace WindowsFormsApp1

{

public partial class SingUp : Form

{

DBAccess ObjDBAccess = new DBAccess();

public SingUp()

{

InitializeComponent();

}

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

{

String userName = txtName.Text;

String userEmail = txtEmail.Text;

String userPassword = txtPassword.Text;

String userCountry = txtCountry.Text;

if (userName.Equals(""))

{

MessageBox.Show("Please enter your user name");

}

else if (userEmail.Equals(""))

{

MessageBox.Show("Please enter your Email");

}

else if (userPassword.Equals(""))

{

MessageBox.Show("Please enter your Password");

}

else if (userCountry.Equals(""))

{

MessageBox.Show("Please enter your Country");

}

else

{

SqlCommand insertCommand = new SqlCommand("insert into Users(Name,Email,Password,Country) values (@userName, @userEmail, @userPassword,@Country");

insertCommand.Parameters.AddWithValue("@userName", userName);

insertCommand.Parameters.AddWithValue("@userEmail", userEmail);

insertCommand.Parameters.AddWithValue("@userPassword", userPassword);

insertCommand.Parameters.AddWithValue("@userCountry", userCountry);

int row =ObjDBAccess.executeQuery(insertCommand);

if(row == 1)

{

MessageBox.Show("Account is Created Successfully ");

this.Hide();

HomePage home = new HomePage();

home.Show();

}

else

{

MessageBox.Show("Error occured Try Again ");

}

}

}

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

//{

// System.Environment.Exit(0);

// }

}

}

DB – access===================================

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Data;

using System.Data.SqlClient;

namespace WindowsFormsApp1

{

class DBAccess

{

private static SqlConnection connection = new SqlConnection();

private static SqlCommand command = new SqlCommand();

private static SqlDataReader DbReader;

private static SqlDataAdapter adapter = new SqlDataAdapter();

public SqlTransaction DbTran;

private static string strConnString = "Data Source=(local);Initial Catalog=SocialNetwork;Integrated Security=True";

public void createConn()

{

try

{

if (connection.State != ConnectionState.Open)

{

connection.ConnectionString = strConnString;

connection.Open();

}

}

catch (Exception ex)

{

throw ex;

}

}

public void closeConn()

{

connection.Close();

}

public int executeDataAdapter(DataTable tblName, string strSelectSql)

{

try

{

if (connection.State == 0)

{

createConn();

}

adapter.SelectCommand.CommandText = strSelectSql;

adapter.SelectCommand.CommandType = CommandType.Text;

SqlCommandBuilder DbCommandBuilder = new SqlCommandBuilder(adapter);

string insert = DbCommandBuilder.GetInsertCommand().CommandText.ToString();

string update = DbCommandBuilder.GetUpdateCommand().CommandText.ToString();

string delete = DbCommandBuilder.GetDeleteCommand().CommandText.ToString();

return adapter.Update(tblName);

}

catch (Exception ex)

{

throw ex;

}

}

public void readDatathroughAdapter(string query, DataTable tblName)

{

try

{

if (connection.State == ConnectionState.Closed)

{

createConn();

}

command.Connection = connection;

command.CommandText = query;

command.CommandType = CommandType.Text;

adapter = new SqlDataAdapter(command);

adapter.Fill(tblName);

}

catch (Exception ex)

{

throw ex;

}

}

public SqlDataReader readDatathroughReader(string query)

{

//DataReader used to sequentially read data from a data source

SqlDataReader reader;

try

{

if (connection.State == ConnectionState.Closed)

{

createConn();

}

command.Connection = connection;

command.CommandText = query;

command.CommandType = CommandType.Text;

reader = command.ExecuteReader();

return reader;

}

catch (Exception ex)

{

throw ex;

}

}

public int executeQuery(SqlCommand dbCommand)

{

try

{

if (connection.State == 0)

{

createConn();

}

dbCommand.Connection = connection;

dbCommand.CommandType = CommandType.Text;

return dbCommand.ExecuteNonQuery();

}

catch (Exception ex)

{

throw ex;

}

}

}

}