M. Ali. Arslan

19F-0348

Lab#09

using System;

using System.Windows.Forms;

using System.Data.SqlClient;

using System.Data;

namespace DB\_Lab\_9

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

SqlConnection con = new SqlConnection(@"Data Source=DESKTOP-6HP9N5M\SQLEXPRESS01;Initial Catalog=dblab;Integrated Security=True");

public int ID;

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

{

Getstudentrecord();

}

private void Getstudentrecord()

{

SqlCommand cmd = new SqlCommand("Select \* from studentTable", con);

DataTable dt = new DataTable();

con.Open();

SqlDataReader sdr = cmd.ExecuteReader();

dt.Load(sdr);

con.Close();

studentdataGridView.DataSource = dt;

}

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

{

if(Isvalid())

{

SqlCommand cmd = new SqlCommand("Insert into studentTable VALUES (@Name, @Father\_Name, @Age, @Semester)", con);

cmd.CommandType = CommandType.Text;

cmd.Parameters.AddWithValue("@Name", namebox.Text);

cmd.Parameters.AddWithValue("@Father\_Name", fnamebox.Text);

cmd.Parameters.AddWithValue("@Age", agebox.Text);

cmd.Parameters.AddWithValue("@Semester", sembox.Text);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("New Student is added..!", "Saved", MessageBoxButtons.OK, MessageBoxIcon.Information);

Getstudentrecord();

}

}

private bool Isvalid()

{

if(namebox.Text== string.Empty)

{

MessageBox.Show("Student name is required", "Failed", MessageBoxButtons.OK, MessageBoxIcon.Error);

return false;

}

if (fnamebox.Text == string.Empty)

{

MessageBox.Show("Father Name is required", "Failed", MessageBoxButtons.OK, MessageBoxIcon.Error);

return false;

}

if (agebox.Text == string.Empty)

{

MessageBox.Show("Age is required", "Failed", MessageBoxButtons.OK, MessageBoxIcon.Error);

return false;

}

if (sembox.Text == string.Empty)

{

MessageBox.Show("Semester is required", "Failed", MessageBoxButtons.OK, MessageBoxIcon.Error);

return false;

}

return true;

}

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

{

namebox.Clear();

fnamebox.Clear();

agebox.Clear();

sembox.Clear();

namebox.Focus();

}

private void studentdataGridView\_CellClick(object sender, DataGridViewCellEventArgs e)

{

ID = Convert.ToInt32(studentdataGridView.SelectedRows[0].Cells[0].Value);

namebox.Text = studentdataGridView.SelectedRows[0].Cells[1].Value.ToString();

fnamebox.Text = studentdataGridView.SelectedRows[0].Cells[2].Value.ToString();

agebox.Text = studentdataGridView.SelectedRows[0].Cells[3].Value.ToString();

sembox.Text = studentdataGridView.SelectedRows[0].Cells[4].Value.ToString();

}

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

{

if(ID > 0)

{

SqlCommand cmd = new SqlCommand("UPDATE studentTable SET Name = @Name, [Father Name] = @Father\_Name, Age = @Age, Semester = @Semester WHERE ID = @ID", con);

cmd.CommandType = CommandType.Text;

cmd.Parameters.AddWithValue("@Name", namebox.Text);

cmd.Parameters.AddWithValue("@Father\_Name", fnamebox.Text);

cmd.Parameters.AddWithValue("@Age", agebox.Text);

cmd.Parameters.AddWithValue("@Semester", sembox.Text);

cmd.Parameters.AddWithValue("@ID", this.ID);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("Student Record is Updated..!", "Updated", MessageBoxButtons.OK, MessageBoxIcon.Information);

Getstudentrecord();

}

else

{

MessageBox.Show("Please Select a student to update..!", "Select", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

if (ID > 0)

{

SqlCommand cmd = new SqlCommand("DELETE FROM studentTable WHERE ID = @ID", con);

cmd.CommandType = CommandType.Text;

cmd.Parameters.AddWithValue("@ID", this.ID);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("Student Record is Deleted..!", "Updated", MessageBoxButtons.OK, MessageBoxIcon.Information);

Getstudentrecord();

}

else

{

MessageBox.Show("Please Select a student to delete..!", "Delete", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

}

Graphical user interface, application

Description automatically generatedGraphical user interface

Description automatically generated

Graphical user interface

Description automatically generatedGraphical user interface, application

Description automatically generated