**31 C# и MS SQL SERVER**

Задание №1. Выполните этапы создания приложения используя главу 3. Выполните подключения к базе данных использую DataAdapter для вашей СУБД.

Изучите главу 4 пункт 4.3, используя приложение из задания №1 создайте и выполните SQL запросы:

выберите все туры;

удалите тур;

добавьте туриста;

измените туриста.

Листинг программы:

public partial class Form1 : Form

{

public SqlConnection sqlConnection = null;

private SqlCommandBuilder sqlBuilder = null;

private SqlDataAdapter sqlDataAdapter = null;

private DataSet dataSet = null;

private string dbPath =

@"D:\Git projects\C-practice\day31\Day 31\30-1\DBTur\_firm.mdf";

public Form1()

{

InitializeComponent();

}

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

{

this.туристыTableAdapter.Fill(this.dBTur\_firmDataSet2.Туристы);

this.турыTableAdapter.Fill(this.dBTur\_firmDataSet1.Туры);

sqlConnection = new SqlConnection($@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename={dbPath};");

sqlConnection.Open();

}

private void toolStripLabel1\_Click(object sender, EventArgs e)

{

editTableForm editTableForm = new editTableForm();

editTableForm.connection = sqlConnection;

editTableForm.ShowDialog();

}

private void toolStripLabel2\_Click(object sender, EventArgs e)

{

var currentCeil = dataGridView1.SelectedCells;

var id = dataGridView1[0, currentCeil[0].RowIndex].Value;

string query = "DELETE FROM Туры "

+ $"WHERE Код\_тура = {id}";

SqlCommand cmd = new SqlCommand(query, sqlConnection);

cmd.ExecuteNonQuery();

}

private void toolStripLabel3\_Click(object sender, EventArgs e)

{

editTableForm editTableForm = new editTableForm();

editTableForm.connection = sqlConnection;

var currentCeil = dataGridView2.SelectedCells;

editTableForm.textBox1.Text = dataGridView2[1, currentCeil[0].RowIndex].Value.ToString();

editTableForm.textBox2.Text = dataGridView2[2, currentCeil[0].RowIndex].Value.ToString();

editTableForm.textBox3.Text = dataGridView2[3, currentCeil[0].RowIndex].Value.ToString() ;

editTableForm.type = "edit";

editTableForm.id = dataGridView2[0, currentCeil[0].RowIndex].Value.ToString();

editTableForm.ShowDialog();

}

}

public partial class editTableForm : Form

{

public SqlConnection connection = null;

public string type = "";

public string id = "";

public editTableForm()

{

InitializeComponent();

}

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

{

if (type == "edit")

{

Form1 form1 = new Form1();

string name = textBox1.Text;

string surname = textBox2.Text;

string secondName = textBox3.Text;

string query = "Update Туристы " +

$"SET Имя='{name}', Фамилия={surname}, Отчество={secondName} " +

$"WHERE Код\_туриста = {id}";

SqlCommand cmd = new SqlCommand(query, connection);

cmd.ExecuteNonQuery();

this.Close();

type = "";

} else

{

Form1 form1 = new Form1();

string name = textBox1.Text;

string surname = textBox2.Text;

string secondName = textBox3.Text;

string query = "Insert Into Туристы " +

$"(Имя, Фамилия, Отчество) Values('{name}', '{surname}', '{secondName})'";

SqlCommand cmd = new SqlCommand(query, connection);

cmd.ExecuteNonQuery();

this.Close();

}

}

}

Таблица 31.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| surname, name, secondName | DataGridView1, DataGridView2 |

Анализ результатов представлен на рисунке 31.1.

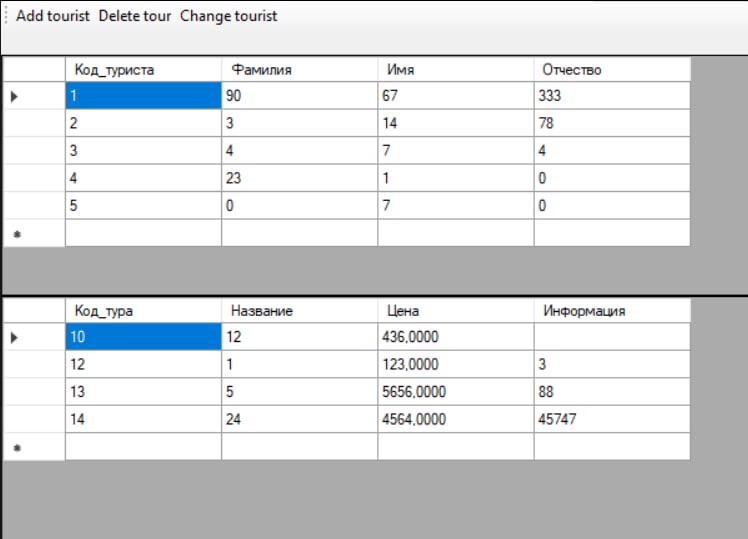


Рисунок 31.1 – Результат работы программы

Источник: собственная разработка