**31 C# и MS SQL Server**

Задание №1. В любой СУБД создайте базу данных DBTur\_firm (Структура и описание таблиц базы данных DBTur\_firm – стр.18, Схема данных базы DBTur\_firm– стр. 21).

Создайте .Net приложение (конкольное WinForm, WPF любое на выбор) и выполните подключение к разработанной базе данных, для этого изучите материал главы 4, пункт 4.1. 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 pathToDB = @"G:\Колледж\Git\C-practice\31\31\z1\DBTur\_firm.mdf";

public Form1()

{

InitializeComponent();

}

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

{

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

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

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

sqlConnection.Open();

}

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

{

EditTable editTable = new EditTable();

editTable.connection = sqlConnection;

editTable.ShowDialog();

}

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

{

var currentCell = dataGridView1.SelectedCells;

var identifier = dataGridView1[0, currentCell[0].RowIndex].Value;

string query = "DELETE FROM Туры "

+ $"WHERE Код\_тура = {identifier}";

SqlCommand comand = new SqlCommand(query, sqlConnection);

comand.ExecuteNonQuery();

}

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

{

EditTable editTable = new EditTable();

editTable.connection = sqlConnection;

var currentCell = dataGridView2.SelectedCells;

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

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

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

editTable.type = "edit";

editTable.identifier = dataGridView2[0, currentCell[0].RowIndex].Value.ToString();

editTable.ShowDialog();

}

}

public partial class EditTable : Form

{

public SqlConnection connection = null;

public string type = "";

public string identifier = "";

public EditTable()

{

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 patronymic = textBox3.Text;

string query = "Update Туристы " +

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

$"WHERE Код\_туриста = {identifier}";

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 patronymic = textBox3.Text

string query = "Insert Into Туристы " +

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

SqlCommand cmd = new SqlCommand(query, connection);

cmd.ExecuteNonQuery();

this.Close();

}

}

}

Таблица 31.1 – Входные и выходные данные

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

Источник: собственная разработка

Анализ результатов представлен на рисунке 31.1.

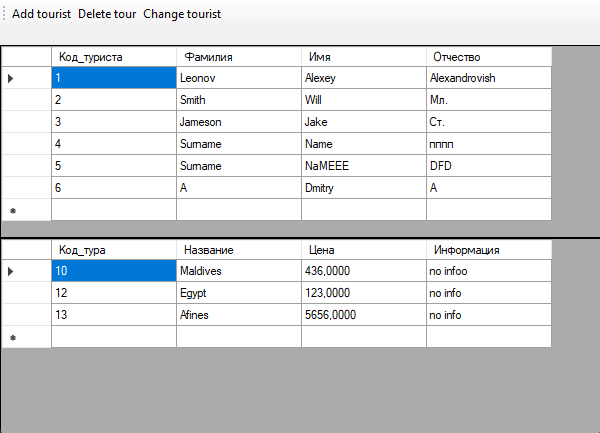


Рисунок 31.1 – Результат работы программы

Источник: собственная разработка