Листинг 1

Program.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace meh2

{

static class Program

{

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

}

// Функция проверки на число армстронга

public static bool isArm(int num) {

int r, sum = 0, temp, l;

l = num.ToString().Length;

temp = num;

while (num > 0) {

r = num % 10;

sum = sum + Convert.ToInt32(Math.Pow(r, l));

num = num / 10;

}

if (temp == sum)

{

return true;

}

else

{

return false;

}

}

}

}

DBAccess.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SQLite;

using System.Data;

using System.IO;

namespace meh2

{ // Класс для работы с БД

public partial class DBAccess: Form1

{

// Переменные названия файлов, строки SQLiteCOnnection, Path, SQLiteCommand

private String dbFileName;

public SQLiteConnection m\_dbConn;

private SQLiteCommand m\_sqlCmd;

private string dbPath;

public DBAccess()

{

}

// При загрузке окна создаем объекты подключения и объект для комманд SQL

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

{

m\_dbConn = new SQLiteConnection();

m\_sqlCmd = new SQLiteCommand();

dbFileName = "DataResult.sqlite";

}

// Функция создания ДБ, если еще не создана и подключение к ней

public void CreateDb()

{

m\_dbConn = new SQLiteConnection();

m\_sqlCmd = new SQLiteCommand();

dbFileName = "DataResult.sqlite";

// Если еще не существует создаем новую ДБ

if (!File.Exists(dbFileName))

SQLiteConnection.CreateFile(dbFileName);

// Оюработка исключений типа SQLiteException

try

{

//Подключение к дб

m\_dbConn = new SQLiteConnection("Data Source=" + dbFileName + ";Version=3;");

m\_dbConn.Open();

m\_sqlCmd.Connection = m\_dbConn;

// Создание двух полей Result и File

m\_sqlCmd.CommandText = "CREATE TABLE IF NOT EXISTS Catalog (Result INTEGER, File TEXT)";

string relativePath = @"Debug\DataResult.sqlite";

var parentdir = Path.GetDirectoryName(Application.StartupPath);

string absolutePath = Path.Combine(parentdir, relativePath);

dbPath = absolutePath;

m\_sqlCmd.ExecuteNonQuery();

}

catch (SQLiteException ex)

{

MessageBox.Show("Error:" + ex.Message);

}

}

// Подключение к Дб если та уже создана

public void connectDb()

{

try

{

//m\_dbConn = new SQLiteConnection("data source=" + "{AppDir}/DATA/" + dbFileName + ";Version=3;");

m\_dbConn = new SQLiteConnection("Data Source=" + dbFileName + ";Version=3;");

m\_dbConn.Open();

m\_sqlCmd.Connection = m\_dbConn;

}

catch (SQLiteException ex)

{

MessageBox.Show("Error: " + ex.Message);

}

}

// Добавление данных результата работы программы(чисел армстронга)

public void addData(int data)

{

// Обработка исключения на не подключеннную ДБ типа SQLiteException

if (m\_dbConn.State != ConnectionState.Open)

{

MessageBox.Show("DB Not Connected!");

return;

}

try

{

m\_sqlCmd.CommandText = "INSERT INTO Catalog('Result') VALUES('"+ data +"')";

m\_sqlCmd.ExecuteNonQuery();

}

catch (SQLiteException ex)

{

MessageBox.Show("Error: " + ex.Message);

}

}

// Добавление пути в поле File

public void addData2()

{

if (m\_dbConn.State != ConnectionState.Open)

{

MessageBox.Show("DB Not Connected!");

return;

}

try

{

m\_sqlCmd.CommandText = "INSERT INTO Catalog('File') VALUES('" + dbPath + "')";

m\_sqlCmd.ExecuteNonQuery();

}

catch (SQLiteException ex)

{

MessageBox.Show("Error: " + ex.Message);

}

}

}

}

Form1.cs

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.SQLite;

using System.Data;

using System.IO;

namespace meh2

{

public partial class Form1 : Form

{

//DBAccess DBAccessV = new DBAccess();

public Form1()

{

InitializeComponent();

}

// Функция срабатывает при нажатии кнопки Calculate("button1")

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

{

var DBAccessV = new DBAccess();

DBAccessV.CreateDb();

DBAccessV.connectDb();

textBox1.Clear();

int n = Convert.ToInt32(textBox2.Text);

getAllNumbers(n, DBAccessV);

DBAccessV.addData2();

}

// Проверко от чисел от 10 до n на то что являются они числами армстронга или нет

// DBAccess - образец класса доступа бд, передается в функцию чтобы оставить уже созданный коннект с БД

private void getAllNumbers(int n, DBAccess DBAccessV)

{

//var DBAccessV = new DBAccess();

for (int i = 10; i < n; i++)

{

if (Program.isArm(i))

{

// Если является записывам в базу

DBAccessV.addData(i);

// Вывод чисел прошедших проверку на экран

if (textBox1.Text.Length > 0)

{

textBox1.AppendText(Environment.NewLine);

}

textBox1.AppendText(i.ToString());

}

}

}

private void richTextBox1\_TextChanged(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

}

}

}

App.config

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

<startup>

<supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.6.1" />

</startup>

</configuration>