

федеральное государственное бюджетное образовательное учреждение высшего образования

«Санкт-Петербургский государственный технологический институт (технический университет)»

**Дисциплина: «Разработка ПС»**

**Отчет по практической работе №4**

**Вариант№9**

Разработка СУБД

«Хранение фильмотеки»

**Выполнил студент группы №475:**

Овчинников Роман Сергеевич

**Проверили:**

Ст. преп. Федин Алексей Константинович.

Доц. Корниенко Иван Григорьевич.

Санкт-Петербург

2019

**1. Постановка задачи:**

Необходимо написать приложение с использованием технологии WinForms для хранения фильмотеки.

**2. Исходные данные**

В качестве исходных данных программа использует данные, вводимые пользователем в поля для ввода.

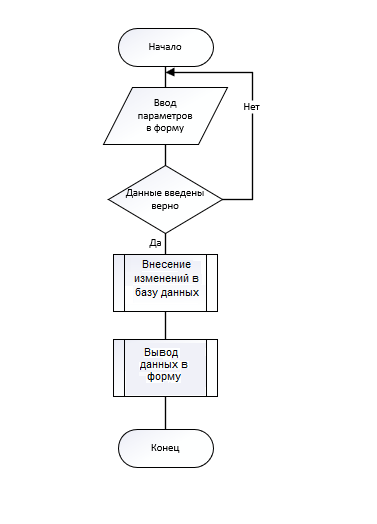
**3. Особые ситуации**

Рассмотреть ситуации, когда были введены символы, не относящиеся к числам, когда не были заполнены поля.

**4. Методы решения задачи**

В форме была создана панель отображения Базы Данных, с помощью которой можно контролировать состояние БД.

**5. Блок-схема**



*Рисунок 1- Блок-схема*

**6. Форматы представленных данных**

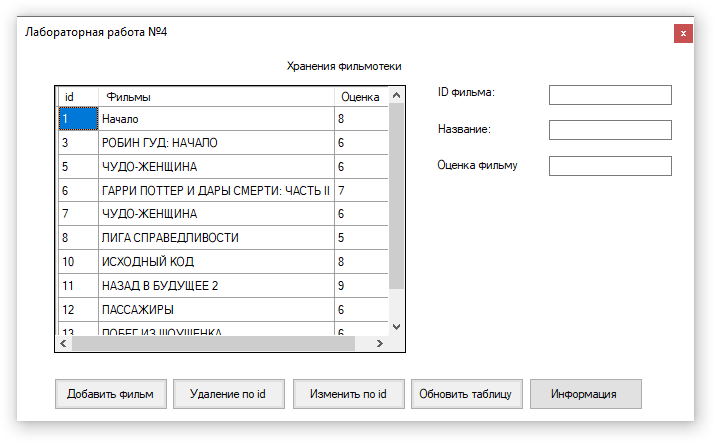
*Таблица 1- Форматы представления данных*

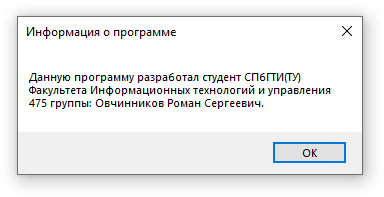
|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Описание** |
| id | var | Первичный ключ |
| tbFILM | varchar | Название фильма |
| tbID | var | Номер зачетка |
| tbAssessment | var | Оценка фильма |

**7. Структура программы**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Описание** |
| btnAdd\_Click | void | Добавление нового фильма |
| btnChange\_Click | void | Изменение данных о фильме по id |
| btnDelete\_Click | void | Удаление фильма по id |
| btnAll\_Click | void | Запрос на отображение текущих данных |

*Таблица 2 - Методы*

**8. Результаты работы программы**

*****Рисунок 2- Результаты работы программы*

*Рисунок 3- Информация о программе*

**9. Исходный текст программы**

[Начало блока Form1.cs]

using System;

using System.Data;

using System.Data.SQLite;

using System.Text.RegularExpressions;

using System.Windows.Forms;

using System.IO;

namespace Lab4

{

public partial class Form1 : Form

{

private SQLiteConnection \_db;

public Form1()

{

InitializeComponent();

}

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

{

\_db = new SQLiteConnection("Data source=FilmsBD.db; Version=3");

\_db.Open();

SQLiteDataAdapter dataAdapter = new SQLiteDataAdapter("select \* from Film", \_db);

DataSet ds = new DataSet();

dataAdapter.Fill(ds, "Info");

grdFilm.DataSource = ds.Tables[0];

}

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

\_db.Close();

}

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

{

if (tbFILM.Text != "" && tbAssessment.Text != "")

{

var allNums = Regex.Matches(tbAssessment.Text, @"\d+");

var assessment = new int[allNums.Count];

for (var i = 0; i < assessment.Length; ++i)

{

assessment[i] = int.Parse(allNums[i].Value);

}

tbAssessment.Text = assessment[0].ToString();

if (1 <= assessment[0] && assessment[0] <= 10)

{

SQLiteCommand cmd = \_db.CreateCommand();

cmd.CommandText = "insert into Film(Фильмы, Оценка) values(@film, @assessment)";

cmd.Parameters.Add("@film", DbType.AnsiString).Value = tbFILM.Text.ToUpper();

cmd.Parameters.Add("@assessment", DbType.Double).Value = assessment[0];

cmd.ExecuteNonQuery();

}

else

{

MessageBox.Show("Оцените фильм от 1 до 10!");

}

btnAll\_Click(sender, e);

}

}

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

{

SQLiteDataAdapter dataAdapter = new SQLiteDataAdapter("select \* from Film", \_db);

DataSet ds = new DataSet();

dataAdapter.Fill(ds, "Info");

grdFilm.DataSource = ds.Tables[0];

}

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

{

var allNums = Regex.Matches(tbID.Text, @"\d+");

var tmp = new int[allNums.Count];

for (var i = 0; i < tmp.Length; ++i)

{

tmp[i] = int.Parse(allNums[i].Value);

}

try

{

tbID.Text = tmp[0].ToString();

SQLiteCommand cmd = \_db.CreateCommand();

try

{

cmd.CommandText = "DELETE FROM Film WHERE id like " + tmp[0];

cmd.ExecuteNonQuery();

btnAll\_Click(sender, e);

}

catch

{

MessageBox.Show(@"Поля не заполнены", @"Обнаружены ошибки");

}

}

catch

{

MessageBox.Show(@"Введите допустимое значение", @"Обнаружены ошибки");

}

}

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

{

var allNums = Regex.Matches(tbID.Text, @"\d+");

var tmp = new int[allNums.Count];

for (var i = 0; i < tmp.Length; ++i)

{

tmp[i] = int.Parse(allNums[i].Value);

}

var allNums2 = Regex.Matches(tbAssessment.Text, @"[-+]?\d+");

var assessment = new int[allNums2.Count];

for (var i = 0; i < assessment.Length; ++i)

{

assessment[i] = int.Parse(allNums2[i].Value);

}

try

{

tbAssessment.Text = assessment[0].ToString();

}

catch

{

tbAssessment.Text = "";

}

try

{

tbID.Text = tmp[0].ToString();

SQLiteCommand cmd = \_db.CreateCommand();

try

{

if (tbFILM.Text != "" && tbAssessment.Text != "")

{

cmd.CommandText = "update Film set Оценка = " + assessment[0] + ", Фильмы = '" +

tbFILM.Text.ToUpper() + "' where id like " + tmp[0];

cmd.ExecuteNonQuery();

if (1 <= assessment[0] && assessment[0] <= 10)

{

cmd = \_db.CreateCommand();

cmd.CommandText = "insert into Film(Фильмы, Оценка) values(@film, @assessment)";

cmd.Parameters.Add("@film", DbType.AnsiString).Value = tbFILM.Text.ToUpper();

cmd.Parameters.Add("@assessment", DbType.Double).Value = assessment[0];

cmd.ExecuteNonQuery();

}

else

{

MessageBox.Show("Оцените фильм от 1 до 10!");

}

btnAll\_Click(sender, e);

}

else if (tbAssessment.Text != "")

{

cmd.CommandText = "update Film set Оценка = " + assessment[0] + " where id like " +

tmp[0];

cmd.ExecuteNonQuery();

if (1 <= assessment[0] && assessment[0] <= 10)

{

cmd = \_db.CreateCommand();

cmd.CommandText = "insert into Film(Фильмы, Оценка) values(@film, @assessment)";

cmd.Parameters.Add("@film", DbType.AnsiString).Value = tbFILM.Text.ToUpper();

cmd.Parameters.Add("@assessment", DbType.Double).Value = assessment[0];

cmd.ExecuteNonQuery();

}

else

{

MessageBox.Show("Оцените фильм от 1 до 10!");

}

}

else if (tbFILM.Text != "")

{

cmd.CommandText = "update Film set Фильмы = '" + tbFILM.Text.ToUpper() +

"' where id like " + tmp[0];

cmd.ExecuteNonQuery();

btnAll\_Click(sender, e);

}

else

{

MessageBox.Show(@"Поля не заполнены", @"Обнаружены ошибки");

}

}

catch

{

MessageBox.Show(@"Введите допустимое значение", @"Обнаружены ошибки");

}

}

catch

{

tbID.Text = "";

tbAssessment.Text = "";

MessageBox.Show(@"Введите допустимое значение", @"Обнаружены ошибки");

}

}

private void grdStud\_RowHeaderMouseClick\_1(object sender, DataGridViewCellMouseEventArgs e)

{

if (grdFilm.Rows[e.RowIndex].Cells[0].Value.ToString() != string.Empty)

{

tbID.Text = grdFilm.Rows[e.RowIndex].Cells[0].Value.ToString();

tbFILM.Text = grdFilm.Rows[e.RowIndex].Cells[1].Value.ToString();

tbAssessment.Text = grdFilm.Rows[e.RowIndex].Cells[2].Value.ToString();

}

}

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

{

MessageBox.Show("Данную программу разработал студент СПбГТИ(ТУ)\nФакультета Информационных технологий и управления\n475 группы: Овчинников Роман Сергеевич.", "Информация о программе");

}

}

}

[Конец блока Form1.cs]