МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «Гомельский государственный технический

университет имени П.О. Сухого»

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

Отчёт по лабораторной работе №8

По дисциплине «Объектно-ориентированное программирование»

**«****Технологии доступа к базам данных»**

Выполнил:

Студент группы ИТП-21

Коркуц С. И.

Принял:

ассистент Карабчикова Е.А.

Гомель 2020

**Цель работы:** изучить технологию доступа к базам данным.

**Задание:**

Реализовать CRUD операции используя технологию LINQ to SQL. В этой лабораторной работе запрещено использовать компоненты.

• Выделенный курсивом параметр расширить до справочной таблицы.

• БД содержать не менее 3х таблиц.

• Обязательно использовать подгрузку данных в элементы управления из справочных таблиц (Например: выпадающие списки).

• Провести модульное тестирование созданных библиотек.

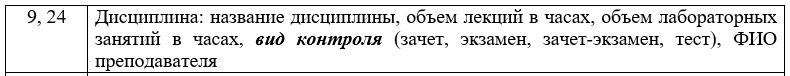


Рисунок 1 – Вариант задания

На рисунке 2 изображена структура решения.

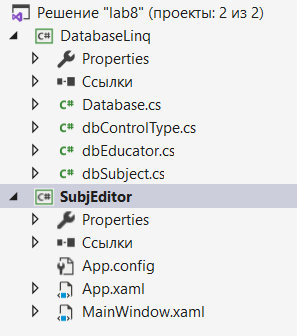


Рисунок 2 – Структура решения

На рисунке 3 изображена схема базы данных.

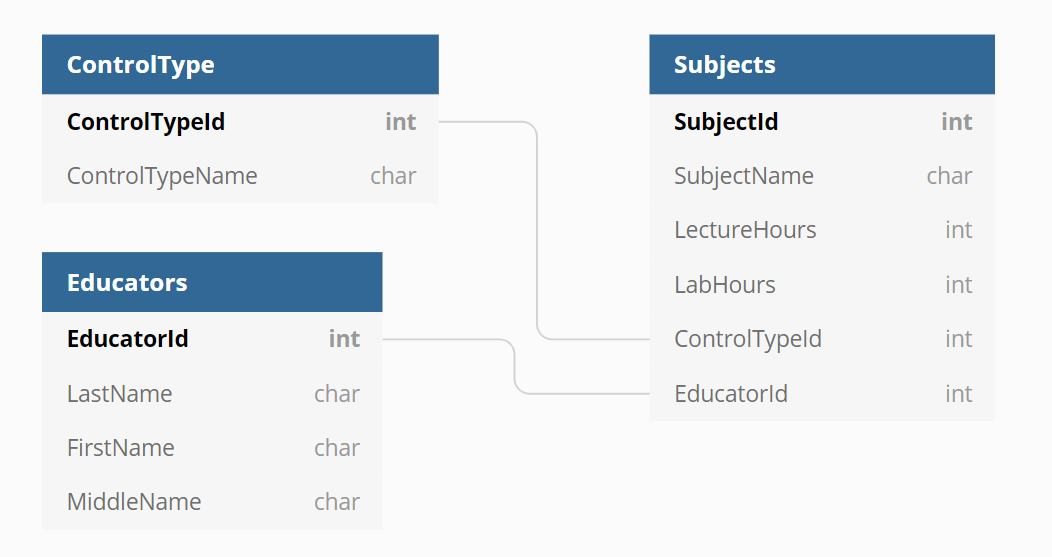


Рисунок 3 – Схема базы данных

При запуске программы открывается основное меню (Рисунок 4).

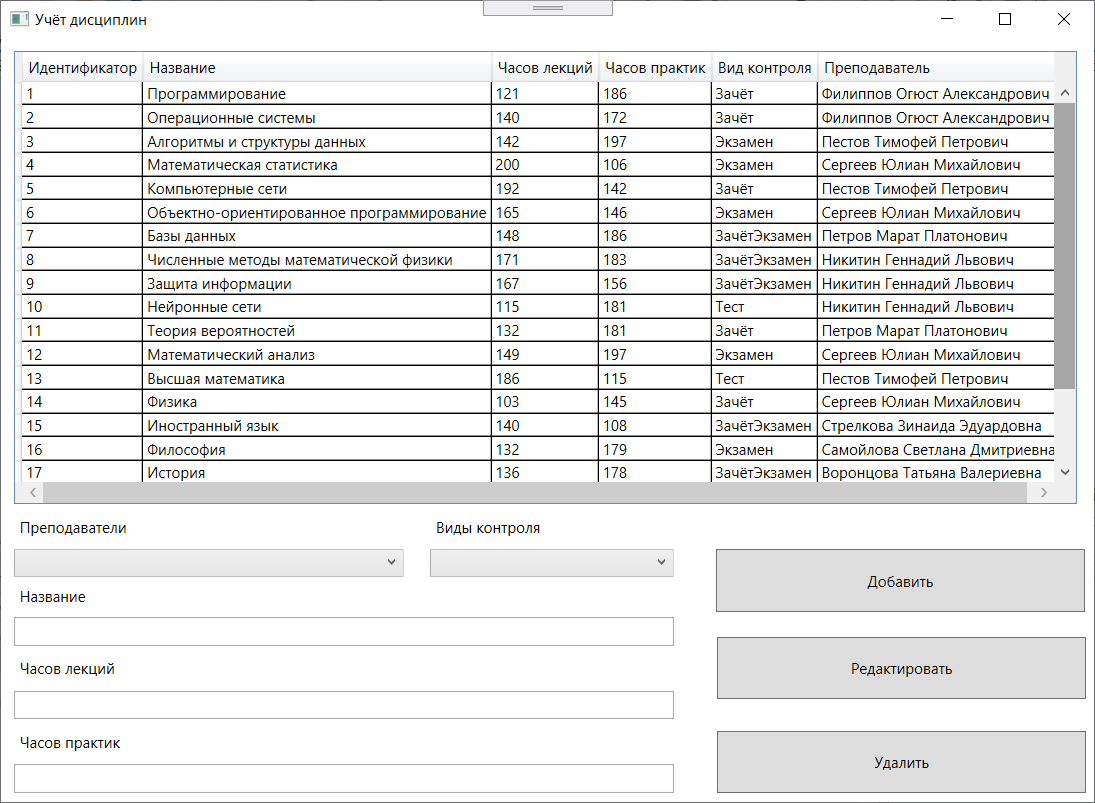


Рисунок 4 – Вывод основного меню

При нажатии кнопки добавления записи, добавляется новая запись в программе и БД (Рисунок 5).

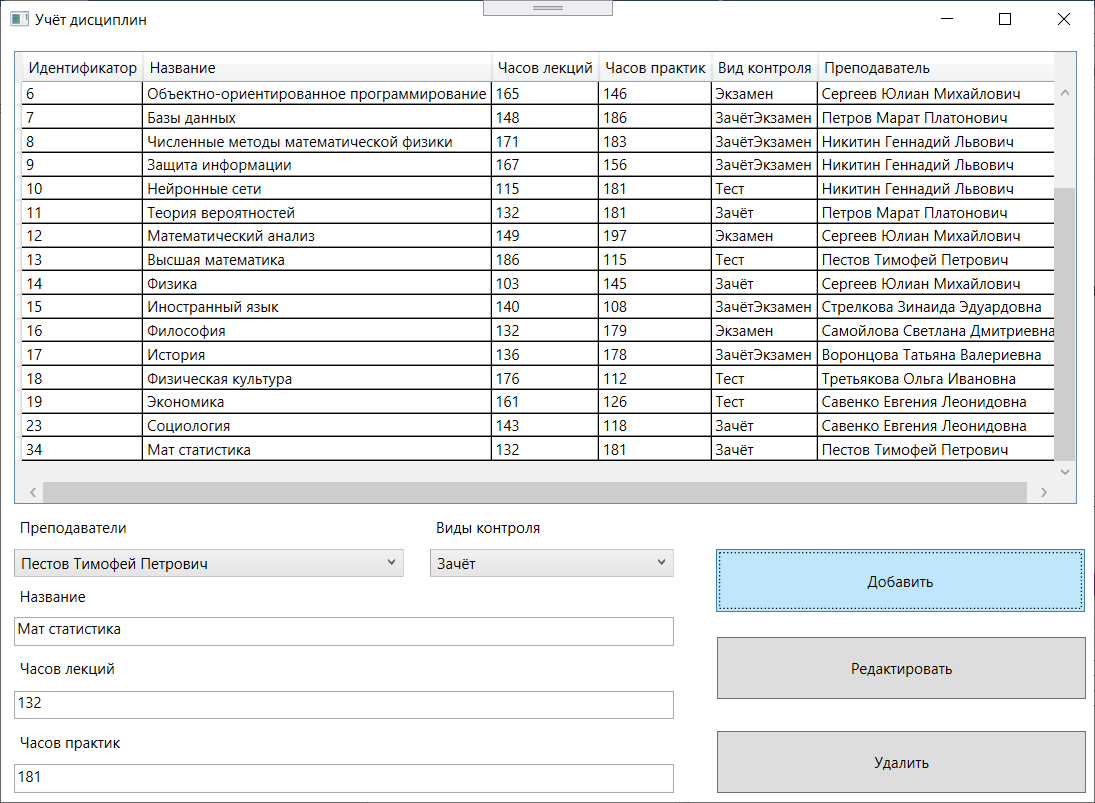


Рисунок 5 – Результат добавления записи

При нажатии кнопки удаления, выбранная запись удаляется из БД (Рисунок 6).

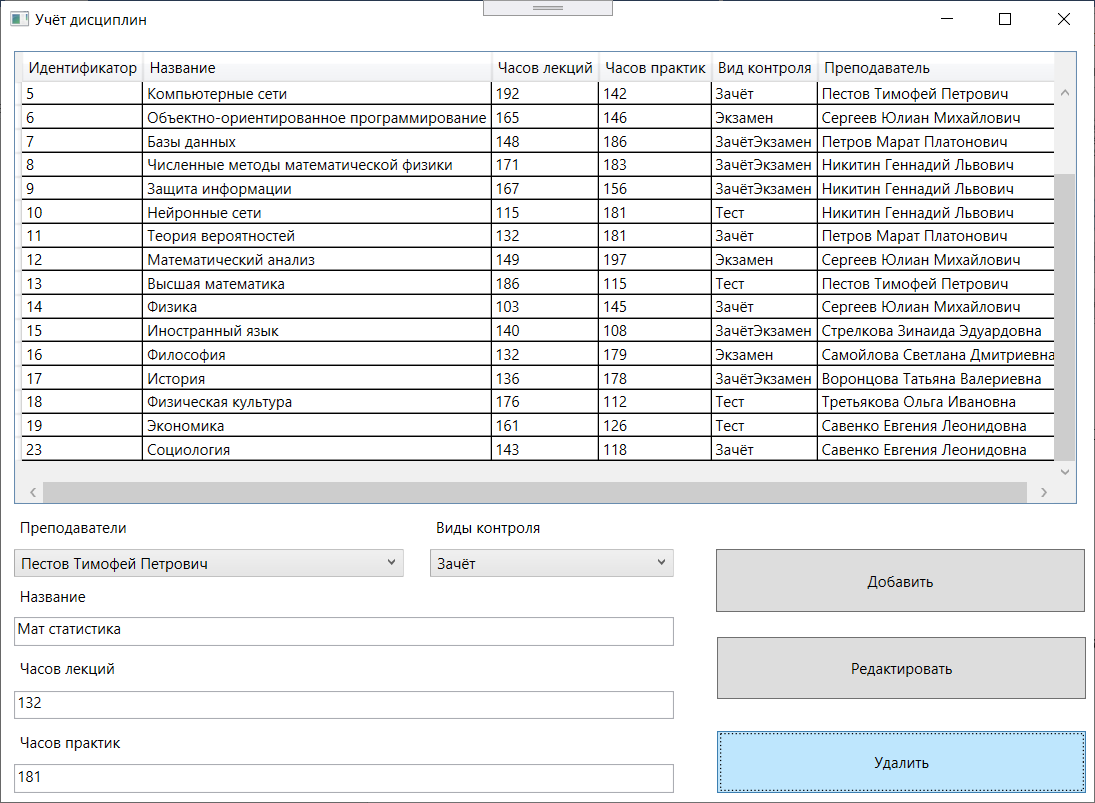


Рисунок 6 – Результат удаления элемента записи

**Вывод:** изучена технология доступа к базам данным.

**ПРИЛОЖЕНИЕ А**

Листинг класса *Database*.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Data;

using System.Data.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DatabaseLinq

{

public static class Database

{

static string connectionString = @"Data Source=.\SQLEXPRESS;Initial Catalog=SubjectsDB;Integrated Security=True";

static DataContext db;

static Table<dbControlType> dbControlTypes;

static Table<dbEducator> dbEducators;

static Table<dbSubject> dbSubjects;

static Database()

{

db = new DataContext(connectionString);

dbControlTypes = db.GetTable<dbControlType>();

dbEducators = db.GetTable<dbEducator>();

dbSubjects = db.GetTable<dbSubject>();

}

public static IQueryable GetControlTypes()

{

dbControlTypes = db.GetTable<dbControlType>();

return from cntrl in dbControlTypes

select new

{

cntrl.ControlTypeId,

ControlTypeName = cntrl.ControlTypeName.TrimEnd(' ')

};

}

public static IQueryable GetEducators()

{

dbEducators = db.GetTable<dbEducator>();

return from edu in dbEducators

select new

{

edu.EducatorId,

EducatorName = string.Join(" ", new string[] { edu.LastName.TrimEnd(' '), edu.FirstName.TrimEnd(' '), edu.MiddleName.TrimEnd(' ') })

};

}

public static IQueryable GetSubjects()

{

dbSubjects = db.GetTable<dbSubject>();

IQueryable rows = from subj in dbSubjects

join cntrl in dbControlTypes on subj.ControlTypeId equals cntrl.ControlTypeId

join edu in dbEducators on subj.EducatorId equals edu.EducatorId

select new

{

subj.SubjectId,

SubjectName = subj.SubjectName.TrimEnd(' '),

subj.LectureHours,

subj.LabHours,

ControlTypeName = cntrl.ControlTypeName.TrimEnd(' '),

Educator = string.Join(" ", new string[] { edu.LastName.TrimEnd(' '), edu.FirstName.TrimEnd(' '), edu.MiddleName.TrimEnd(' ') })

};

return rows;

}

public static void AddSubject(string subjectName, int lecHours, int labHours, int controlTypeId, int educatorId)

{

dbSubject subject = new dbSubject();

subject.SubjectName = subjectName;

subject.LectureHours = lecHours;

subject.LabHours = labHours;

subject.ControlTypeId = controlTypeId;

subject.EducatorId = educatorId;

dbSubjects.InsertOnSubmit(subject);

db.SubmitChanges();

}

public static void EditSubject(int id, string subjectName, int lecHours, int labHours, int controlTypeId, int educatorId)

{

var subjects = from subj in dbSubjects

where subj.SubjectId == id

select subj;

foreach (var subj in subjects)

{

subj.SubjectName = subjectName;

subj.LectureHours = lecHours;

subj.LabHours = labHours;

subj.ControlTypeId = controlTypeId;

subj.EducatorId = educatorId;

}

db.SubmitChanges();

}

public static void DeleteSubject(int id)

{

var subjects = from subj in dbSubjects

where subj.SubjectId == id

select subj;

foreach (var subj in subjects)

{

dbSubjects.DeleteOnSubmit(subj);

}

db.SubmitChanges();

}

}

}

**ПРИЛОЖЕНИЕ Б**

Листинг класса *MainWindow.xaml*.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Data;

using DatabaseLinq;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace SubjEditor

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

InitializeDataGrid();

InitializeEducatorsBox();

InitializeControlBox();

}

void InitializeDataGrid()

{

subjectsGrid.Items.Clear();

var subjects = Database.GetSubjects();

foreach (var subject in subjects)

{

subjectsGrid.Items.Add(subject);

}

}

void InitializeEducatorsBox()

{

var educators = Database.GetEducators();

educatorsBox.SelectedValuePath = "Key";

educatorsBox.DisplayMemberPath = "Value";

foreach (var edu in educators)

{

educatorsBox.Items.Add(new KeyValuePair<int, string>((int)edu.GetType().GetProperty("EducatorId").GetValue(edu), (string)edu.GetType().GetProperty("EducatorName").GetValue(edu)));

}

}

void InitializeControlBox()

{

var controls = Database.GetControlTypes();

controlBox.SelectedValuePath = "Key";

controlBox.DisplayMemberPath = "Value";

foreach (var cntrl in controls)

{

controlBox.Items.Add(new KeyValuePair<int, string>((int)cntrl.GetType().GetProperty("ControlTypeId").GetValue(cntrl), (string)cntrl.GetType().GetProperty("ControlTypeName").GetValue(cntrl)));

}

}

private void subjectsGrid\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

try

{

educatorsBox.Text = subjectsGrid.SelectedItem.GetType().GetProperty("Educator").GetValue(subjectsGrid.SelectedItem).ToString();

controlBox.Text = subjectsGrid.SelectedItem.GetType().GetProperty("ControlTypeName").GetValue(subjectsGrid.SelectedItem).ToString();

subjectNameBox.Text = subjectsGrid.SelectedItem.GetType().GetProperty("SubjectName").GetValue(subjectsGrid.SelectedItem).ToString();

lectureHoursBox.Text = subjectsGrid.SelectedItem.GetType().GetProperty("LectureHours").GetValue(subjectsGrid.SelectedItem).ToString();

labHoursBox.Text = subjectsGrid.SelectedItem.GetType().GetProperty("LabHours").GetValue(subjectsGrid.SelectedItem).ToString();

}

catch

{

}

}

private void addButton\_Click(object sender, RoutedEventArgs e)

{

Database.AddSubject(subjectNameBox.Text, int.Parse(lectureHoursBox.Text), int.Parse(labHoursBox.Text), (int)controlBox.SelectedItem.GetType().GetProperty("Key").GetValue(controlBox.SelectedItem), (int)educatorsBox.SelectedItem.GetType().GetProperty("Key").GetValue(educatorsBox.SelectedItem));

InitializeDataGrid();

}

private void editButton\_Click(object sender, RoutedEventArgs e)

{

try

{

Database.EditSubject((int)subjectsGrid.SelectedItem.GetType().GetProperty("SubjectId").GetValue(subjectsGrid.SelectedItem), subjectNameBox.Text, int.Parse(lectureHoursBox.Text), int.Parse(labHoursBox.Text), (int)controlBox.SelectedItem.GetType().GetProperty("Key").GetValue(controlBox.SelectedItem), (int)educatorsBox.SelectedItem.GetType().GetProperty("Key").GetValue(educatorsBox.SelectedItem));

InitializeDataGrid();

}

catch

{

MessageBox.Show("Выберите поле для редактирования");

}

}

private void deleteButton\_Click(object sender, RoutedEventArgs e)

{

try

{

Database.DeleteSubject((int)subjectsGrid.SelectedItem.GetType().GetProperty("SubjectId").GetValue(subjectsGrid.SelectedItem));

InitializeDataGrid();

}

catch

{

MessageBox.Show("Выберите поле для удаления");

}

}

}

}

**ПРИЛОЖЕНИЕ В**

Листинг классов таблиц.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Data.Linq.Mapping;

using System.Text;

using System.Threading.Tasks;

namespace DatabaseLinq

{

[Table(Name = "ControlTypes")]

public class dbControlType

{

[Column(IsPrimaryKey = true, IsDbGenerated = true)]

public int ControlTypeId { get; set; }

[Column]

public string ControlTypeName { get; set; }

}

}

namespace DatabaseLinq

{

[Table(Name = "Educators")]

public class dbEducator

{

[Column(IsPrimaryKey = true, IsDbGenerated = true)]

public int EducatorId { get; set; }

[Column]

public string LastName { get; set; }

[Column]

public string FirstName { get; set; }

[Column]

public string MiddleName { get; set; }

}

}

namespace DatabaseLinq

{

[Table(Name = "Subjects")]

public class dbSubject

{

[Column(IsPrimaryKey = true, IsDbGenerated = true)]

public int SubjectId { get; set; }

[Column]

public string SubjectName { get; set; }

[Column]

public int LectureHours { get; set; }

[Column]

public int LabHours { get; set; }

[Column]

public int ControlTypeId { get; set; }

[Column]

public int EducatorId { get; set; }

}

}