МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение   
высшего образования

**«МОСКОВСКИЙ АВТОМОБИЛЬНО–ДОРОЖНЫЙ   
ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ (МАДИ)»**

Факультет «Управление»

Кафедра «Автоматизированные системы управления»

09.04.01 – «Интегрированные автоматизированные системы управления в отраслях транспортно–дорожного комплекса»

Дисциплина: «Технология разработки программного обеспечения*»*

**Отчет**

по лабораторной (самостоятельной) работе

на тему:

«**Разработка информационной системы для колы чародейства и волшебства “Хогвартс”»**

Выполнил:

Проверил:

**Москва  
2023**

Термины и сокращения

|  |  |  |
| --- | --- | --- |
| БД | – | База данных |
| СУБД | – | Система управления базами данных |
| ПО | – | Программное обеспечение |
| ИС | – | Информационная система |

Оглавление

[Диаграмма базы данных 4](#_Toc155717921)

[Пользовательский интерфейс 5](#_Toc155717922)

[Удаление студента из списка школы 6](#_Toc155717923)

[Изменение информации о студенте 7](#_Toc155717924)

[Добавление нового студента 9](#_Toc155717925)

[Программная реализация 12](#_Toc155717926)

[Техническая информация 12](#_Toc155717927)

[Программный код 12](#_Toc155717928)

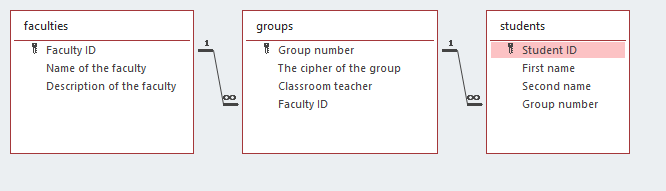
[Программный код для основной формы 12](#_Toc155717929)

[Программный код для дополнительной формы 14](#_Toc155717930)

# Диаграмма базы данных

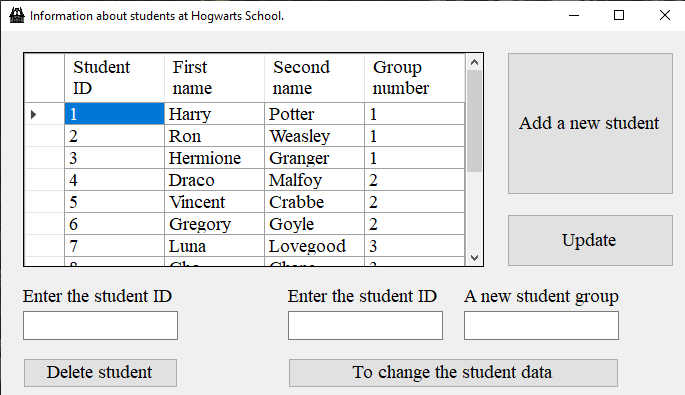
Так как в школе чародейства и волшебства Хогвартс не будет учиться большое количество студентов и школьников, то в качестве основной системы управления базами данных (далее СУБД) был выбран MS Access. Так как СУБД MS Access позволяет реализовать реляционные базы данных (далее БД), то в дальнейшем можно будет осуществить миграцию БД под другую СУБД, например MS SQL.

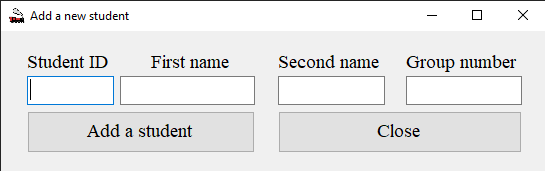
На рис. 1 представлена диаграмма базы данных.

  
Рисунок 1. Диаграмма базы данных

# Пользовательский интерфейс

Так как программа подразумевает использование на стационарном компьютере, то пользовательский интерфейс состоит из одной основной формы (рис. 2) и одной вспомогательной формы (рис. 3).

  
Рисунок 2. Основная форма приложения

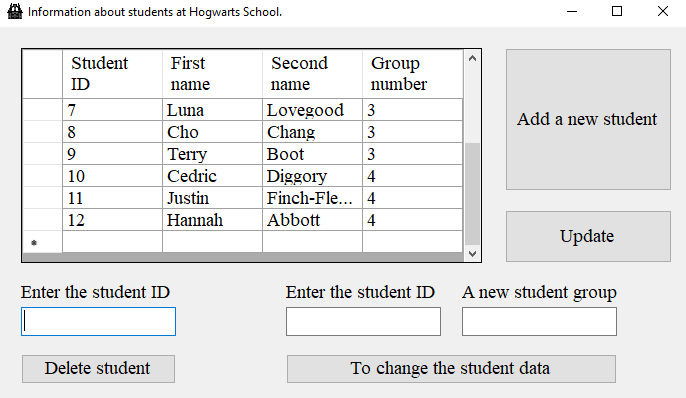
  
Рисунок 3. Вспомогательная форма

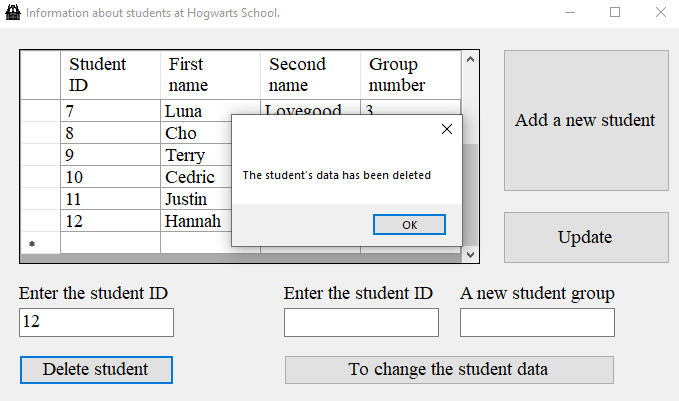
Основная форма предназначена для учебного отдела и отражает список всех студентов с указанием группы. Из главной формы есть возможность удалить студента из списка школы, изменить номер группы, вызвать вспомогательную форму для добавления нового студента[[1]](#footnote-1).

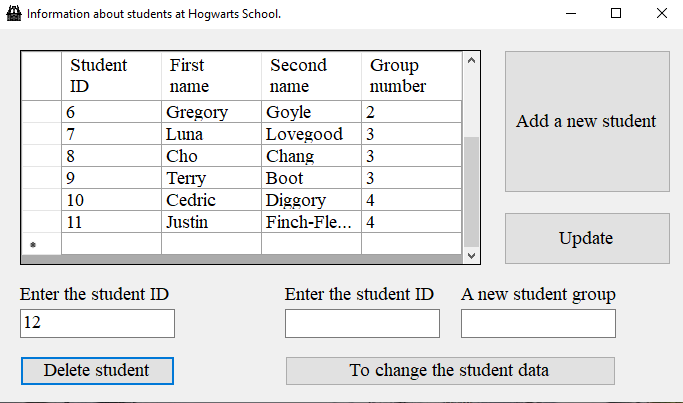
## Удаление студента из списка школы

Для того чтобы удалить студента необходимо проделать следующие шаги:

1. В крайнем нижнем текстовом поле ввести номер студента (рис. 4);
2. Нажить кнопку «Delete student» (рис. 5);
3. Если операция удаления прошла успешно, то появиться окно, уведомляющее об успешности операции, в котором необходимо нажать на кнопку «ОК» (рис. 5). В результате будет удалён студент, ID которого было введено для удаления (рис. 6).

  
Рисунок 4. Выбор поля для удаления студента школы

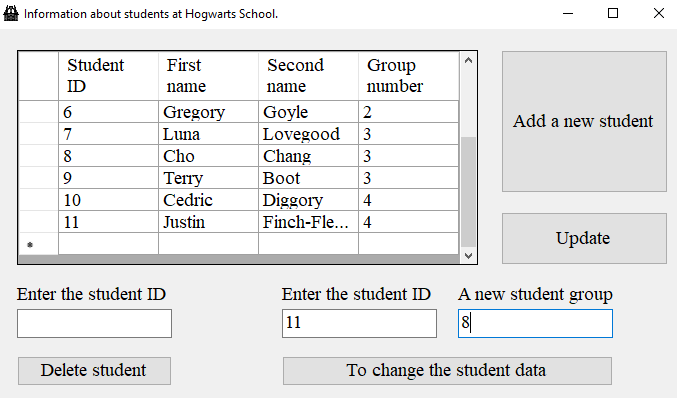
  
Рисунок 5. Ввод в поле ID школьника и окно уведомлений вызванное нажатием на кнопку «Delete students»

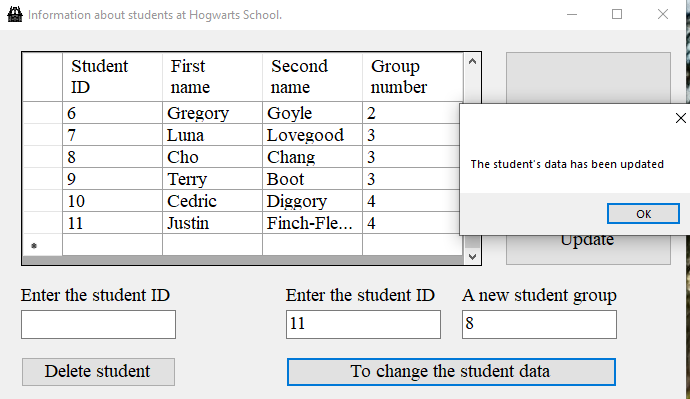
  
Рисунок 6. Результат выполнения удаления студента

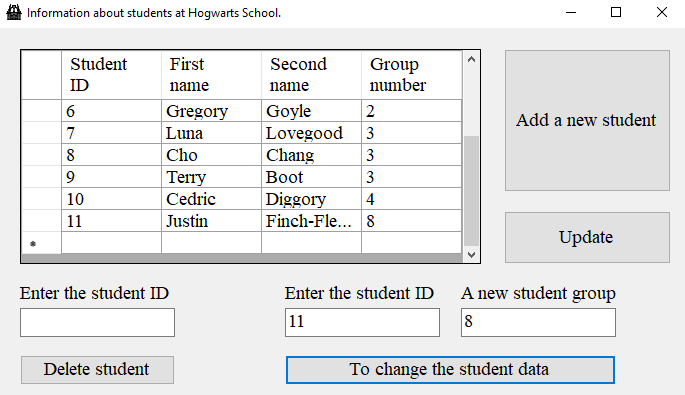
## Изменение информации о студенте

Для изменения информации о студенте необходимо проделать следующие шаги:

1. В правой нижней части основной формы в текстовых полях введи ID–студента, для которого необходимо изменить информацию, и обновлённый номер группы (рис. 7);
2. Нажать кнопку «To change the student data», при корректно введённых данных всплывёт диалоговое окно, которое сообщит об успешности обновления данных (рис. 8);
3. В информационном окне нажимаем кнопку «ОК» и данные будут автоматически обновлены (рис. 9).

  
Рисунок 7. Ввод информации для изменения

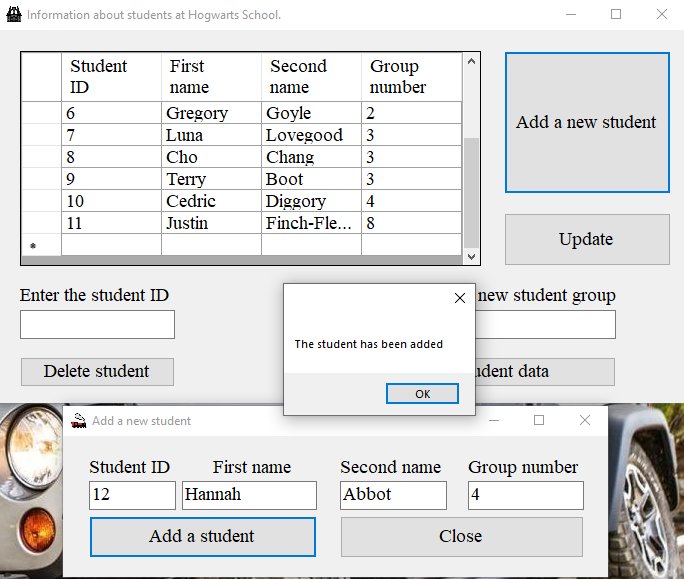
  
Рисунок 8. Тестовое сообщение, коотрое выводиться при успешном выполнении функции кнопки «To change the student data»

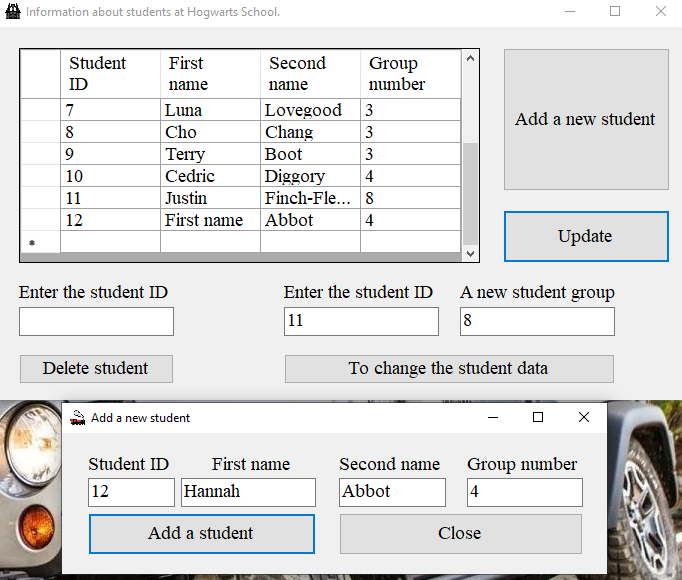
  
Рисунок 9. Итог обновления данных

## Добавление нового студента

Для добавление нового студента в список школы необходимо проделать следующие шаги:

1. Нажать кнопку «Add a new student» (рис. 10);
2. В появившейся форме в текстовых полях ввести данные студента (рис. 10), такие как:
   1. ID;
   2. Фамилия студента;
   3. Имя студента;
   4. Номер группы;
3. Нажать кнопку «add student» (рис. 10);
4. После того как программа выведет текстовое сообщение об успешном завершение запроса, в текстовом окне надо нажать кнопку «OK», а форме добавления студента нажать кнопку «Close» (рис. 10);
5. В основной форме информационной системы нажать кнопку «Update» (рис. 11).

  
Рисунок 10. Ввод информации о новом студенте

  
Рисунок 11. Обновление таблицы основной формы

# Программная реализация

## Техническая информация

Приложение написано на языке C#, запросы к базе данных на языке T–SQL. Требования к программной реализации представлены в табл. 1.

Таблица . Технические требования к языкам программирования

|  |  |
| --- | --- |
| Язык программирования приложения | .NETFramework 4.7 |
| Язык запросов к Базе данных | oledb.12.0 (универсальное решение для запросов на языке SQL и T–SQL) |

Для обработки запросов к базе данных была использована технология oledb, которая позволяет формировать запросы к базам данных как на языке SQL[[2]](#footnote-2), так и на языке T–SQL[[3]](#footnote-3).

## Программный код

### Программный код для основной формы

using System;

using System.Windows.Forms;

using System.Data.OleDb;

namespace HarryPotter

{

    public partial class Form1 : Form

    {

        public static string connectionString = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=|DataDirectory|\\HarryPotter2002–2003.mdb";

        private OleDbConnection myString;

        public Form1()

        {

            InitializeComponent();

            myString = new OleDbConnection(connectionString);

            myString.Open();

        }

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

        {

            this.studentsTableAdapter.Fill(this.\_HarryPotter2002\_2003DataSet.students);

        }

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

        {

            myString.Close();

        }

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

        {

            int deleteIDstudents = Convert.ToInt32(textBoxDelete.Text);

            string deleteQuery = "DELETE FROM students WHERE [Student ID] = " + deleteIDstudents;

            OleDbCommand command = new OleDbCommand(deleteQuery, myString);

            command.ExecuteNonQuery();

            MessageBox.Show("The student's data has been deleted");

            this.studentsTableAdapter.Fill(this.\_HarryPotter2002\_2003DataSet.students);

        }

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

        {

            int changeIDstudents = Convert.ToInt32(textBoxSelectionStudent.Text);

            int changeGroup = Convert.ToInt32(textBoxChange.Text);

            if (changeGroup > 0 && changeGroup < 21)

            {

                string changeQuery = "UPDATE students SET [Group number] = " + changeGroup + " WHERE [Student ID] = " + changeIDstudents;

                OleDbCommand command = new OleDbCommand(changeQuery, myString);

                command.ExecuteNonQuery();

                MessageBox.Show("The student's data has been updated");

                this.studentsTableAdapter.Fill(this.\_HarryPotter2002\_2003DataSet.students);

            }

            else

            {

                MessageBox.Show("You entered an incorrect group number");

            }

        }

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

        {

            Form2 f2 = new Form2();

            f2.Owner = this;

            f2.Show();

        }

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

        {

            this.studentsTableAdapter.Fill(this.\_HarryPotter2002\_2003DataSet.students);

        }

    }

}

### Программный код для дополнительной формы

using System;

using System.Data.OleDb;

using System.Windows.Forms;

namespace HarryPotter

{

    public partial class Form2 : Form

    {

        public static string connectionString = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=|DataDirectory|\\HarryPotter2002–2003.mdb";

        private OleDbConnection myString;

        public Form2()

        {

            InitializeComponent();

            myString = new OleDbConnection(connectionString);

            myString.Open();

        }

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

        {

            Close();

        }

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

        {

            int StudentID = Convert.ToInt32(textBoxStudentID.Text);

            string FirstName = textBox2.Text;

            string SecondName = textBoxSecondName.Text;

            int GroupNumber = Convert.ToInt32(textBoxGroupNumber.Text);

            string addQuery = "INSERT INTO students VALUES (" + StudentID + ",'" + FirstName + "','" + SecondName + "'," + GroupNumber +")";

            OleDbCommand command = new OleDbCommand(addQuery, myString);

            command.ExecuteNonQuery();

            MessageBox.Show("The student has been added");

        }

    }

}

# Список литературы

**CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#.** / авт. Джеффри Рихтер. – Санкт‑Петербург: Питер, 2022.

**Проектирование баз данных** / авт. Суркова Н. Е. Исмоилов М.И. – Москва: МАДИ, 2012.

**Справочник C#. Кратко, быстро, под рукой** / авт. Дубовик Е. В. Евдокимов П. В. – Москва: Наука и техника, 2023.

1. Кнопка **update** применяется для обновления основной формы после добавления нового студента. [↑](#footnote-ref-1)
2. SQL (Structure Query Language) – язык запросов, применяемый для работы с базами данных. [↑](#footnote-ref-2)
3. T–SQL (Transact Structure Query Language) – процедурное расширение для языка SQL, разработанное компанией Microsoft для MS SQL Server и Sybase. [↑](#footnote-ref-3)