**Лабораторная работа №8**

**Задание:**

Одна запись в журнале зимней экзаменационной сессии представляет собой структуру с полями: курс, код группы, фамилия студента, номер зачетной книжки, дисциплина, оценка за экзамен по дисциплине. Вычисляются средние баллы по дисциплине, группе, курсу. Организовать поиск по курсу, группе, номеру зачетной книжки, фамилии, оценкам.

**Ход работы**:

Создадим класс Record для обработки одной записи журнала экзаменационной сессии.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Xml.Linq;

namespace Gulyaev\_AG\_8

{

[Serializable]

public class Record

{

public ushort Year { get; set; }

public string Group { get; set; }

public string LastName { get; set; }

public ulong RecordBook { get; set; }

public string Subject { get; set; }

public ushort Grade { get; set; }

public Record() { }

public Record(ushort Year, string Group, string LastName, ulong RecordBook, string Subject, ushort Grade)

{

this.Year = Year;

this.Group = Group;

this.LastName = LastName;

this.RecordBook = RecordBook;

this.Subject = Subject;

this.Grade = Grade;

}

public Record(Record magazine)

{

this.Year = magazine.Year;

this.Group = magazine.Group;

this.LastName = magazine.LastName;

this.RecordBook = magazine.RecordBook;

this.Subject = magazine.Subject;

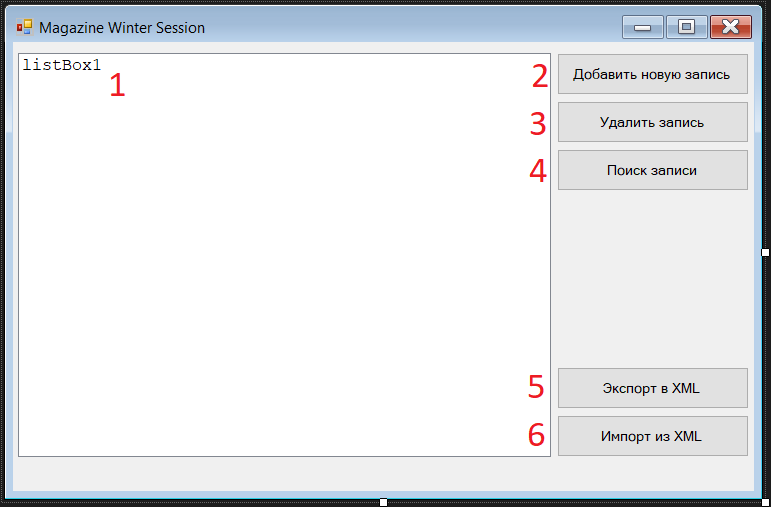
this.Grade = magazine.Grade;

}

}

}

Создадим форму, которая предусматривает вывод записей из xml файла (1), добавление новой записи (2), удаление новой записи (3), поиск записи (4), сохранение текущего отчета в xml-файл (5) и загрузка отчета из xml-файла (6).



В форме объявим два поля

public static bool AddingCanceled = false;

static public List<Record> magazine = new List<Record>();

AddingCanceled – вспомогательный флаг при добавлении записей.

Magazine – список записей, подгружаемых из файла.

Организуем импорт из XML. Для этого на форму добавляем openFileDialog и обработаем нажатие на кнопку (6).

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

{

if (openFileDialog1.ShowDialog() == DialogResult.Cancel) return;

string filename = openFileDialog1.FileName;

SelectFile(filename);

listBox1.Items.Clear();

foreach (Record rec in magazine)

listBox1.Items.Add(rec.Year + " " + rec.Group + " " + rec.LastName + " " + rec.RecordBook + " " + rec.Subject + " " + rec.Grade);

}

Из openFileDialog мы получаем полный путь до файла. Метод SelectFile загружает этот файл, последовательно обходит его узлы, формирует из них запись и заполняет magazine. После обновления содержимого списка magazine мы чистим листбокс от устаревших данных и выводим новые данные.

public static void SelectFile(string filename)

{

int errorInXml = 0;

int cnt\_rec = 1;

magazine.Clear();

try

{

XmlDocument xDoc = new XmlDocument();

errorInXml = 7;

xDoc.Load(filename);

// Корневой элемент

XmlElement xRoot = xDoc.DocumentElement;

// обход всех узлов в корневом элементе

foreach (XmlNode xnode in xRoot)

{

string Element = "";

Record record = new Record();

// обход всех дочерних узлов элемента

foreach (XmlNode childnode in xnode.ChildNodes)

{

// year

if (childnode.Name.ToLower() == "year")

{

errorInXml = 1;

record.Year = UInt16.Parse(childnode.InnerText);

if (record.Year > 6) record.Year = 6;

if (record.Year == 0) record.Year = 1;

}

// group

if (childnode.Name.ToLower() == "group")

{

errorInXml = 2;

record.Group = childnode.InnerText;

record.Group = record.Group.Replace(" ", string.Empty);

}

// lastName

if (childnode.Name.ToLower() == "lastname")

{

errorInXml = 3;

record.LastName = childnode.InnerText;

record.LastName = record.LastName.Replace(" ", string.Empty);

}

// recordBook

if (childnode.Name.ToLower() == "recordbook")

{

errorInXml = 4;

record.RecordBook = UInt64.Parse(childnode.InnerText);

if (record.RecordBook == 0) record.RecordBook = 1;

}

// subject

if (childnode.Name.ToLower() == "subject")

{

errorInXml = 5;

record.Subject = childnode.InnerText;

record.Subject = record.Subject.Replace(" ", string.Empty);

}

// grade

if (childnode.Name.ToLower() == "grade")

{

errorInXml = 6;

record.Grade = UInt16.Parse(childnode.InnerText);

if (record.Grade > 5) record.Grade = 5;

if (record.Grade < 2) record.Grade = 2;

}

}

cnt\_rec++;

magazine.Add(record);

}

}

catch (Exception ex)

{

switch (errorInXml)

{

case 0: MessageBox.Show(filename + " не найден"); break;

case 1: MessageBox.Show("В записи " + cnt\_rec + " некорректное поле year"); break;

case 2: MessageBox.Show("В записи " + cnt\_rec + " некорректное поле group"); break;

case 3: MessageBox.Show("В записи " + cnt\_rec + " некорректное поле lastName"); break;

case 4: MessageBox.Show("В записи " + cnt\_rec + " некорректное поле recordBook"); break;

case 5: MessageBox.Show("В записи " + cnt\_rec + " некорректное поле subject"); break;

case 6: MessageBox.Show("В записи " + cnt\_rec + " некорректное поле grade"); break;

case 7: MessageBox.Show(ex.Message); break;

default: MessageBox.Show(ex.Message); errorInXml = 8; break;

}

MessageBox.Show("Загрузка последующих записей остановлена");

}

}

Организуем экспорт в XML. На форму добавляем saveFileDialog и обрабатываем нажатие на кнопку (5).

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

{

if (saveFileDialog1.ShowDialog() == DialogResult.Cancel) return;

string filename = saveFileDialog1.FileName;

XmlSerializer formatter = new XmlSerializer(typeof(List<Record>));

using (FileStream fs = new FileStream(filename, FileMode.Create))

{

formatter.Serialize(fs, magazine);

MessageBox.Show("Файл успешно сохранен");

}

}

Мы получаем полный путь сохранения (включая новое имя файла) и сериализуем каждую запись нашего журнала.

Организуем удаление выбранной записи. Обработаем нажатие на кнопку (3).

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

{

try {

magazine.RemoveAt(listBox1.SelectedIndex);

listBox1.Items.RemoveAt(listBox1.SelectedIndex);

}

catch (Exception ex)

{

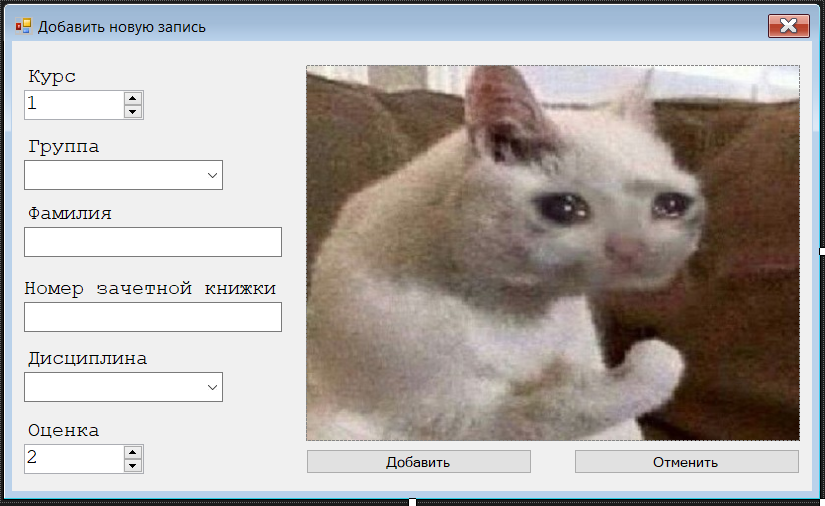
return;

}

}

Мы удаляем из журнала элемент по индексу выбранной в листбоксе записи.

Организуем добавление новой записи. Для этого создадим новую форму FormAdd.



Эта форма предусматривает:

* выбор курса от 1 до 6
* выбор группы из списка групп. Список формируется из файла Groups.txt. Если в xml файле будет группа, которой не будет в group.txt, соответствующая запись будет существовать в программе, но не будет выполняться поиск по этой группе
* ввод фамилии
* ввод номера зачетной книжки (только цифрами)
* выбор дисциплины из списка дисциплин. Список формируется из файла Subjects.txt. Работает аналогично списку групп
* выбор оценки от 2 до 5

Если добавление элемента отменяется, то выставляем значение параметра AddingCanceled из Form1 true и закрываем форму.

При добавлении записи мы удаляем граничные пробелы у имени и номера зачетной книжки, формируем новую запись и добавляем её в magazine.

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

{

textBoxLastName.Text.Trim(' ');

textBoxRecordbook.Text.Trim(' ');

if (textBoxLastName.Text != "" || textBoxRecordbook.Text != "")

{

Record record = new Record(UInt16.Parse(numericUpDownYear.Value.ToString()), comboBoxGroup.Text, textBoxLastName.Text, UInt64.Parse(textBoxRecordbook.Text), comboBoxSubject.Text, UInt16.Parse(numericUpDownGrade.Value.ToString()));

Form1.magazine.Add(record);

Form1.AddingCanceled = false;

this.Close();

}

else { MessageBox.Show("Некорректное заполнение полей."); }

}

В Form1 обработаем нажатие на кнопку (2). Если добавление записи не было отменено, то добавленная запись выводится в листбокс.

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

{

FormAdd formAdd = new FormAdd();

formAdd.ShowDialog();

if (!AddingCanceled)

{

Record rec = magazine.Last();

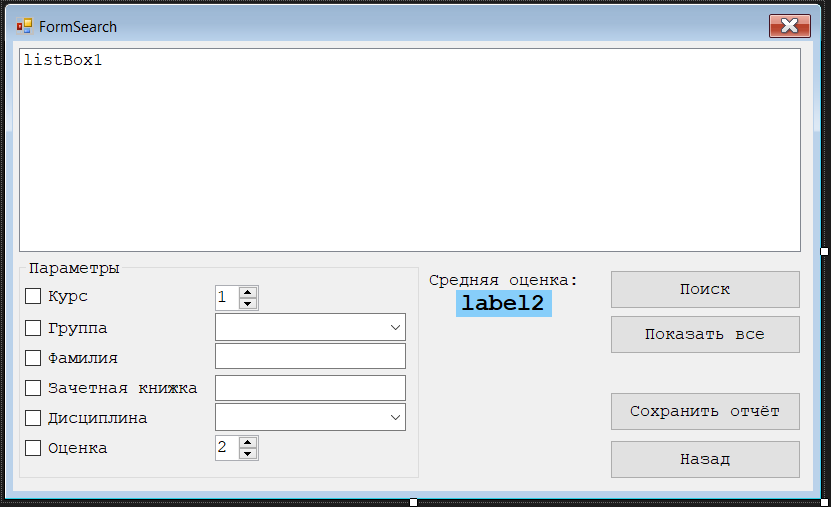
listBox1.Items.Add(rec.Year + " " + rec.Group + " " + rec.LastName + " " + rec.RecordBook + " " + rec.Subject + " " + rec.Grade);

}

AddingCanceled = false;

}

Организуем поиск записи. Создадим новую форму FormSearch. В листбоксе выводятся данные выборки, которая проводится по выбранным параметрам. По каждой из выборок высчитывается средняя оценка. Каждую из выборок можно сохранить в xml.



При загрузке форма получает журнал записей, с которым и работает в дальнейшем. Отбор записей проходит в список записей SaveRecords. В comboBox-ы из файлов Groups.txt и Subjects.txt подгружаются соответствующие записи.

Выборка записей осуществляется в методе MakeSelection следующим образом: заносим в SaveRecords все записи из Magazine. Затем последовательно проходимся по каждому из чекбоксов. Если чекбокс активен, то выбираем записи, соответствующие введенным данным, и выбранные данные перезаписываем в SaveRecords.

List<Record> SearchRecords = new List<Record>();

try

{

listBox1.Items.Clear();

SaveRecords.Clear();

SearchRecords.AddRange(magazine);

if (checkBoxLastName.Checked && textBoxLastName.Text != "")

{

IEnumerable<Record> evens = from i in SearchRecords

where i.LastName.Equals(textBoxLastName.Text)

select i;

SearchRecords = new List<Record>(evens);

}

if (checkBoxRecordbook.Checked && textBoxRecordbook.Text != "")

{

IEnumerable<Record> evens = from i in SearchRecords

where i.RecordBook.Equals(UInt64.Parse(textBoxRecordbook.Text))

select i;

SearchRecords = new List<Record>(evens);

}

if (checkBoxYear.Checked)

{

IEnumerable<Record> evens = from i in SearchRecords

where i.Year.Equals(ushort.Parse(numericUpDownYear.Value.ToString()))

select i;

SearchRecords = new List<Record>(evens);

}

if (checkBoxGroup.Checked)

{

IEnumerable<Record> evens = from i in SearchRecords

where i.Group.Equals(comboBoxGroup.Text)

select i;

SearchRecords = new List<Record>(evens);

}

if (checkBoxSubject.Checked)

{

IEnumerable<Record> evens = from i in SearchRecords

where i.Subject.Equals(comboBoxSubject.Text)

select i;

SearchRecords = new List<Record>(evens);

}

if (checkBoxGrade.Checked)

{

IEnumerable<Record> evens = from i in SearchRecords

where i.Grade.Equals(UInt16.Parse(numericUpDownGrade.Value.ToString()))

select i;

Record rec = SearchRecords.First();

SearchRecords = new List<Record>(evens);

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

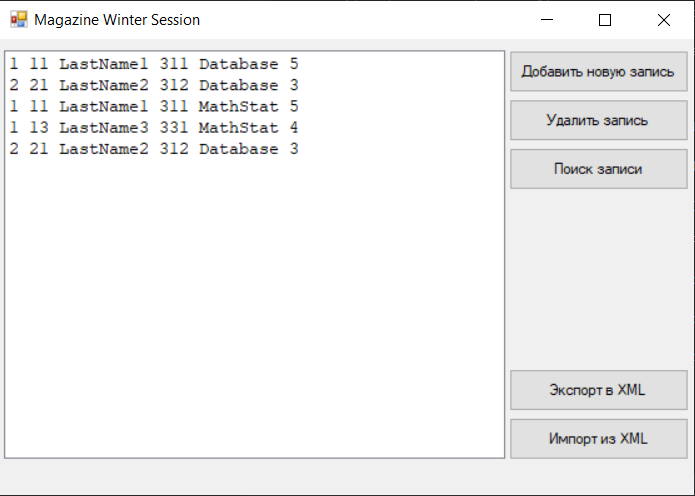
}

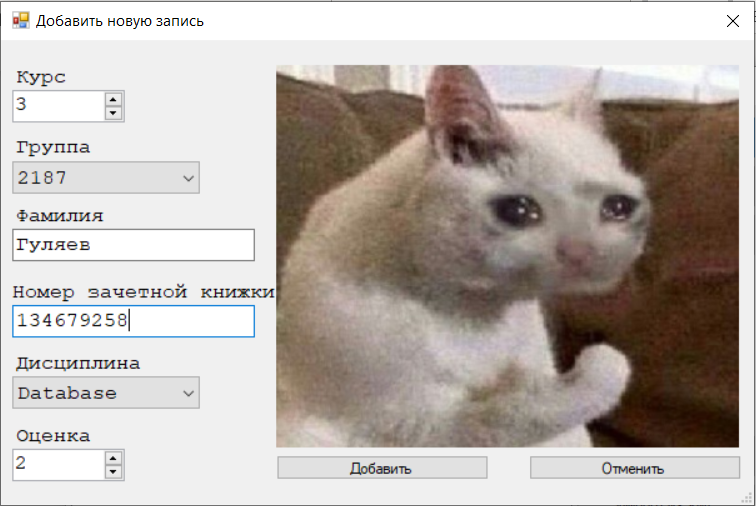
return SearchRecords;

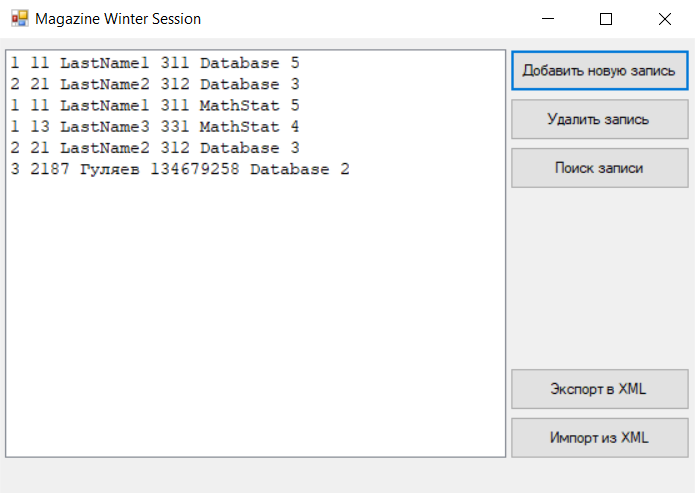
Подсчёт среднего происходит при нажатии кнопок «Поиск» и «Показать всё».

**Демонстрация**:

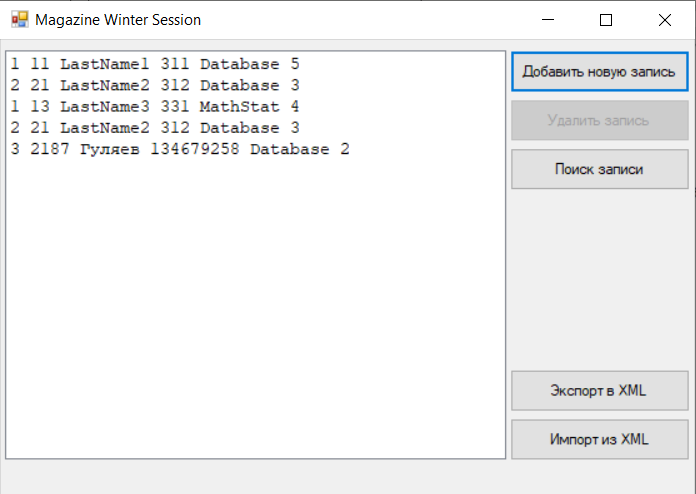
Добавим новую запись



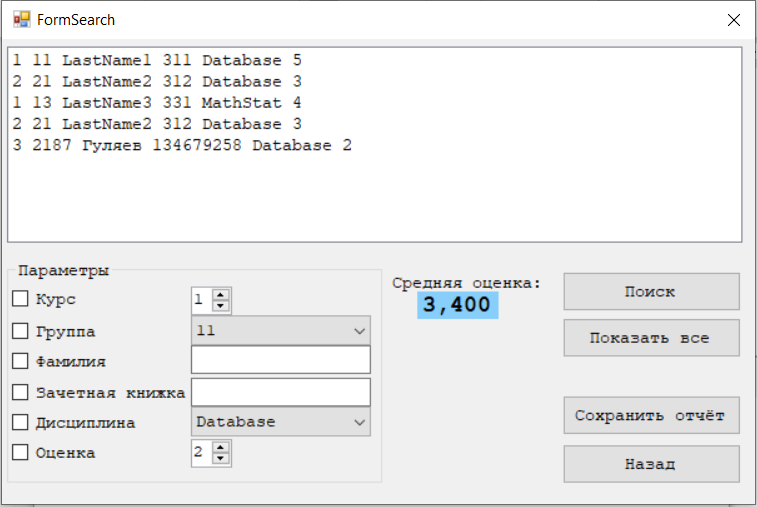




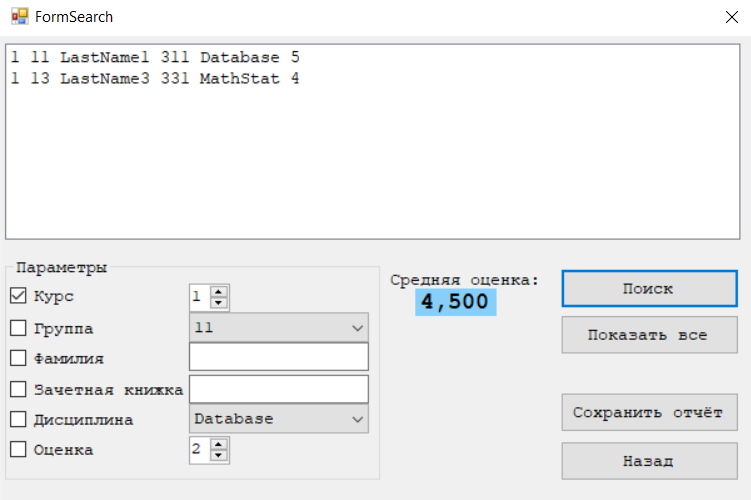
Удалим третью запись



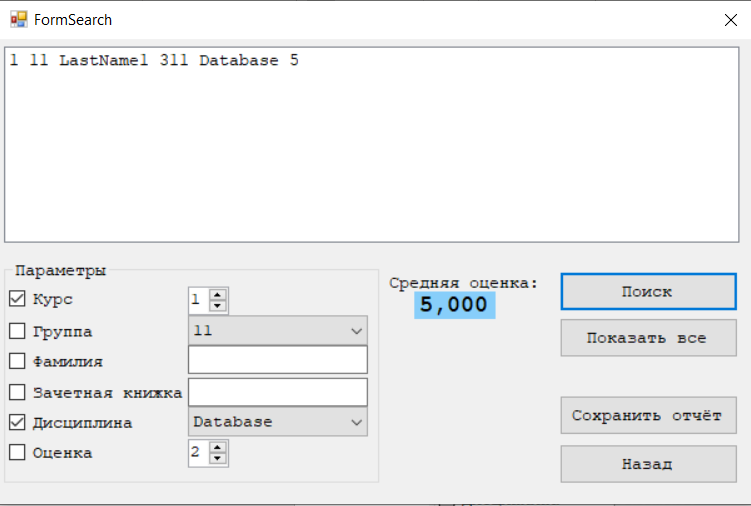
Поиск записей



Сгруппируем данные, например, по первому курсу



Сгруппируем по первому курсу и дисциплине Database.



Сгруппируем по первому курсу, дисциплине Database и оценке 4.

