**Содержание**

[Введение 4](#_Toc420347646)

[1 Исследование предметной области 5](#_Toc420347647)

[2 Функциональная модель информационной системы 7](#_Toc420347648)

[3 Проектная часть 9](#_Toc420347649)

[3.1 Описание методов 9](#_Toc420347650)

[3.2 Описание работы программы 14](#_Toc420347651)

[Заключение 24](#_Toc420347652)

[Список литературы 25](#_Toc420347653)

Введение

Цель курсовой работы: разработка информационной системы в IDE Embarcadero Delphi «Зачетная книжка студента». Информационная система должна представлять собой стандартное windows-приложение с реализацией следующих основных функций: создание записи, добавление записи, сохранение в файл, открытие файла, редактирование, поиск, диаграммы. Для хранения информации использовать базу данных или текстовый файл.

Задачи:

1. исследовать предметную область;
2. составить информационную модель;
3. реализовать графический интерфейс;
4. разработать функциональную часть программы;
5. отладить работу программы.

1 Исследование предметной области

Информационная система представляет собой модель зачетной книжки студента.

Зачетная книжка показывает успеваемость студента и содержит записи, содержащие информацию о сданных отчетностях. Отчетностями являются дифференцированный и недифференцированный зачеты, экзамены и курсовые работы. Каждая запись содержит следующие поля: № записи, тип записи, наименование дисциплины, количество зачетных единиц, оценка, дата сдачи экзамена, фамилия преподавателя, номер семестра.

Информационная система позволяет построить диаграмму, позволяющую оценить общую успеваемость студента.

Физическая модель информационной системы будет реализована следующим образом:

Таблица 1 – Физическая модель ИС

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Название | Тип данных | Размер, байт |
| Number | Номер | Byte | 1 |
| SubType | Тип | string | 50 |
| Subject | Наименование дисциплины | String | 50 |
| Count | Количество з/е | String | 10 |
| Mark | Оценка | String | 30 |
| Data | Дата | TDateTime | 1 |
| Surname | Фамилия преподавателя | String | 30 |
| Semester | Семестр | Byte | 1 |

В качестве графической иллюстрации используется диаграмма, показывающая результаты всех зачетных единиц в процентном соотношении:

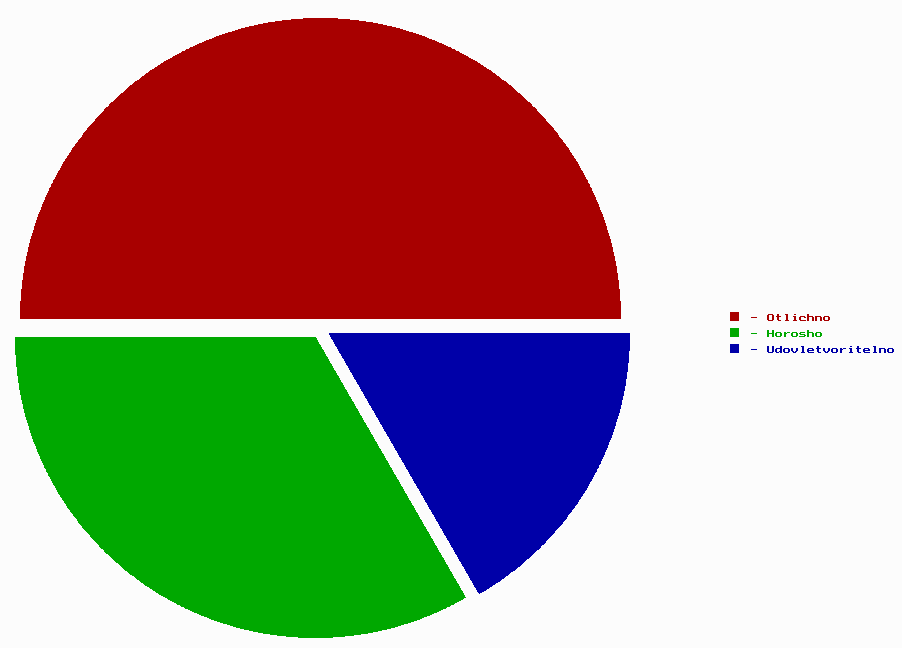


Рисунок 1 - Графическая иллюстрация

Графическая сущность представляет собой следующую схему:



Рисунок 2 - Графическая сущность

2 Функциональная модель информационной системы

Таблица 2 – Физическая модель ИС

|  |  |  |  |
| --- | --- | --- | --- |
| Функция приложения | Описание функции | Наименование действия | Доступ к действию |
| Добавление записи | В динамический список добавляется новая запись об отчетности | ActionAdd | Главное меню/Панель управления |
| Поиск записи | Программа выводит все записи, удовлетворяющие введенному ключу | ActionFind | Главное меню/Панель управления |
| Редактирование записи | В имеющуюся запись вводятся изменения | ActionEdit | Главное меню/Панель управления |
| Удаление записи | Из списка удаляется 1 запись | ActionDelete | Главное меню/Панель управления |
| Построение диаграммы | Строится диаграмма по результатам экзаменов | ActionDiagram | Главное меню/Панель управления |
| Изменение цвета фона | Меняется цвет фона в отображающей области | ActionBackground Color | Пункт в главном меню |
| Изменение цвета шрифта | Меняется цвет шрифта | ActionFont Color | Пункт в главном меню |
| Изменение размера шрифта | Меняется размер шрифта | ActionFont Size | Пункт в главном меню |
| Изменение шрифта | Меняется шрифт | ActionFont | Пункт в главном меню |
| Полужирный шрифт | Меняется начертание | ActionBold | Пункт в главном меню |
| Наклонный шрифт | Меняется начертание | ActionItalic | Пункт в главном меню |
| Полужирный наклонный | Меняется начертание | ActionBoldItalic | Пункт в главном меню |

Таблица 2 – Физическая модель ИС(продолжение)

|  |  |  |  |
| --- | --- | --- | --- |
| Функция приложения | Описание функции | Наименование действия | Доступ к действию |
| Обычный шрифт | Меняется начертание | ActionSimpleFont | Пункт в главном меню |
| Переход на форму «Цвет фона» | Открывается форма с настройками цвета фона | ActionBackground | Главное меню |
| Переход на форму «Шрифт» | Открывается форма с настройками шрифта | ActionFont | Главное меню/Панель управления |
| Вывод информации о программе | Выводится информация о программе | ActionInfo | Главное меню |
| Сохранить документ | Сохраняются изменения в документе | ActionSave | Главное меню/Панель управления |
| Сохранить документ под другим именем | Документ сохраняется под другим именем | ActionSaveAs | Главное меню |
| Уменьшить размер шрифта | Размер шрифта уменьшается на 1 | ActionFontDec | Главное меню/Панель управления |
| Увеличить размер шрифта | Размер шрифта увеличивается на 1 | ActionFontInc | Главное меню/Панель управления |

3 Проектная часть

3.1 Описание методов

Данная процедура выполняет сохранение информации в файл, в качестве параметров передается указатель динамического списка и имя файла.

Procedure TMAinForm.Saving(First: pItem; Name: string);

var F: file of tdata;

begin

Сперва выполняется связывание файловой переменной с заданным именем файла и открытие файла.

AssignFile(F, Name);

Rewrite(F);

После этого информация записывается файл до тех пор, пока указатель не достигнет конца динамического списка. Файл закрывается.

While First<>nil do begin

write(F, First^.data);

First:=First^.next;

end;

CloseFile(F);

end;

Следующая процедура выполняет открытие файла, в качестве парметров передаются указатель динамического списка и имя файла.

Procedure TMainForm.OpenFile(var First: pITem; Name: string);

var F: File of tdata; i:integer; s:tdata; last, k: pITem;

begin

Создаем вспомогательный динамический список, связываем файл с переменной, открываем файл и выполняем проверку на корректность открытия файла.

New(k);

AssignFile(F, Name);

{$I-}Reset(F);{$I+}

If ioresult<>0 then begin ShowMessage('Файл не найден');

exit;

Считываем 1 элемент файла во вспомогательный список.

end else begin

read(F,s);

k^.data:=s;

k^.next:=first;

first:=k;

Считываем весь файл в основной список и закрываем файл.

while not (eof(f)) do begin

Read(F, s);

AddForm.AddLast(First, s);

end;

CloseFile(F);

end;

end;

Следующий метод выполняет построение диаграммы.

procedure TGraphicForm.FormShow(Sender: TObject);

begin

Очищается контейнер Series1.

Series1.Clear;

Выполняется загрузка результатов семестров в переменные i,j,k.

DataLoad(First,i,j,k);

На основе 3 переменных строится график.

Series1.Add(i,'Отлично', clYellow);

Series1.Add(j,'Хорошо',clRed);

Series1.Add(k,'Удовлетворительно',clBlue);

//end;

end;

Следующая процедура выполняет редактирование данных, в качестве параметра передается указатель динамического списка.

Procedure TEditForm.EditList(First: pItem);

begin

Сначала находится элемент списка, который необходимо отредактировать.

while (First^.Data.Number<>Trigger^.Data.Number ) do

First:=First^.Next;

Затем происходит присваивание новых значений переменным в элементе списка.

First^.Data.SubType:=Edit1.Text;

First^.Data.subject:=Edit2.Text;

First^.Data.Count:=Edit3.Text;

First^.Data.mark:=Edit4.Text;

First^.Data.Date:=StrToDate(Edit5.Text);

First^.Data.surname:=Edit6.Text;

First^.Data.semester:=StrToInt(Edit7.Text);

end;

Следующий метод выполняет поиск необходимого элемента в списке, в качестве параметра передается указатель динамического списка.

Procedure TFindForm.FindList(First: pItem);

var j: word;

begin

j:=1;

Если пользователь выбрал необходимое ему значение искомого элемента, то выполняется перебор элементов динамического списка. При наличии искомого элемента запись выводится в таблицу.

if Combobox1.Text<>'' then

begin

while First<>Nil do begin

if First^.Data.SubType = Combobox1.Text then

begin

FindSG.Cells[0,j]:=IntToStr(First^.Data.Number);

FindSG.Cells[1,j]:=First.Data.SubType;

FindSG.Cells[2,j]:=First.Data.subject;

FindSG.Cells[3,j]:=First.Data.Count;

FindSG.Cells[4,j]:=First.Data.mark;

FindSG.Cells[5,j]:=DateToStr(First.Data.Date);

FindSG.Cells[6,j]:=First.Data.surname;

FindSG.Cells[7,j]:=IntToStr(First.Data.semester);

inc(j);

First:=First^.Next;

end

else

First:=First^.Next;

Если во всем списке искомого элемента не существует, то выводится сообщение о том, что искомый элемент не найден.

if (FindSG.Cells[0,1]='') and (First = Nil) then

begin

ShowMessage('Запись не найдена');

exit;

end;

end;

end;

end;

3.2 Описание работы программы

Информационная система представляет собой стандартное windows-приложение, где реализованы следующие возможности:

Пользователь может создать новую запись. Для выполнения этого действия пользователь должен нажать кнопку «создать запись», откроется форма добавления новой записи в динамический список. После этого необходимо ввести данные в текстовые поля и выбрать тип зачетной единицы. Нажать кнопку «Ок». Программа осуществит инициализацию динамического списка и добавление первой записи.

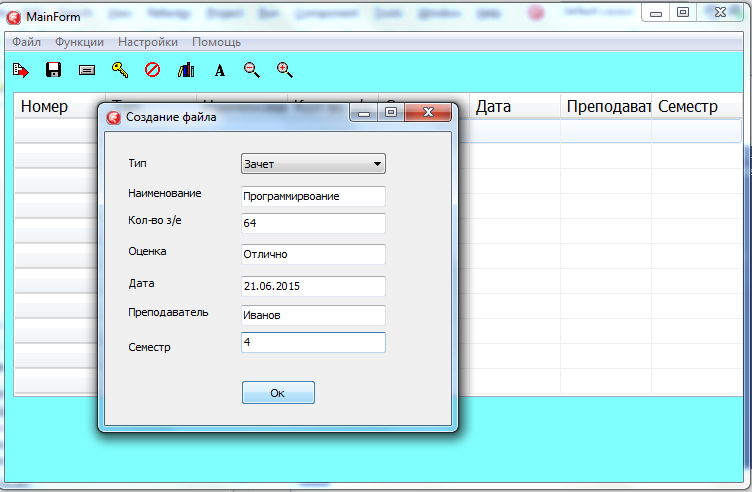


Рисунок 3 - Форма создания новой записи

В случае, если список уже был инициализирован и содержал какие-либо данные, то программа выведет следующее сообщение:

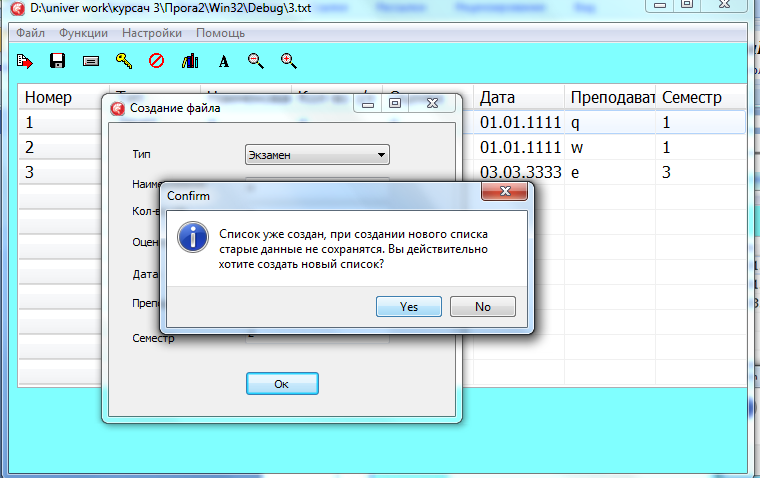


Рисунок 4 - Сообщение о подтверждении действия

При нажатии на кнопку «No» программа отменит действие и закроет форму. При нажатии на кнопку «Yes» программа обнулит старый список.

Возможно добавить в список новую запись. Для этого необходимо выбрать пункт меню «Добавить запись» и откроется форма добавления записи. После этого необходимо ввести данные в текстовые поля и выбрать тип зачетной единицы. Нажать кнопку «Ок». Программа осуществит добавление нового элемента в динамический список. В том случае, если список был пустым, то программа откроет форму создания записи вместо формы добавления новой записи.

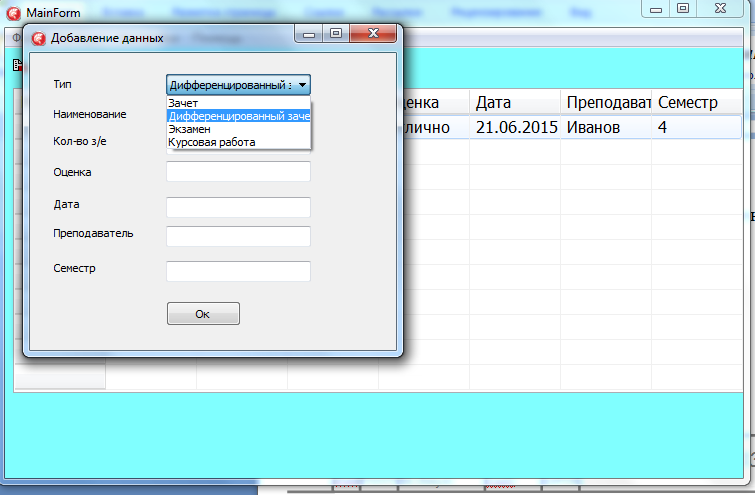


Рисунок 5 - Добавление новой записи

Для сохранения информации в файл необходимо выбрать пункт главного меню «Сохранить» или нажать на соответствующую кнопку на панели инструментов. Если сохраняются изменения в существующем файле, то старый файл перезапишется.

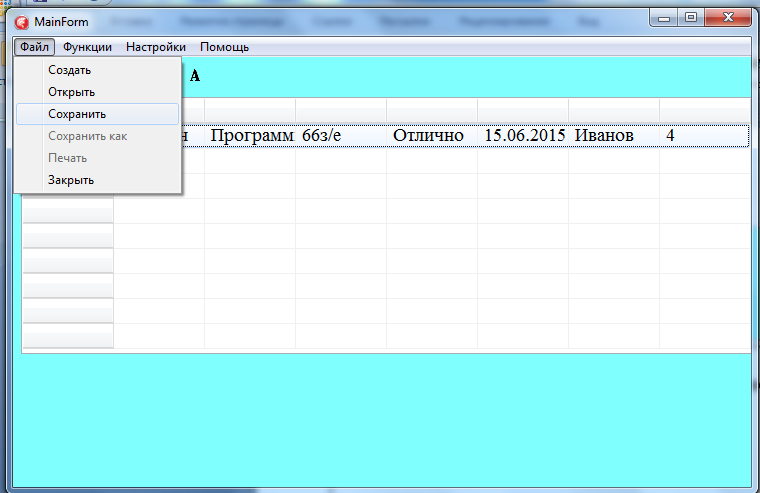


Рисунок 6 - Сохранение информации

Если сохраняются изменения в новом файле, то появится диалоговое окно, в котором необходимо будет ввести имя нового файла и выбрать путь сохранения:

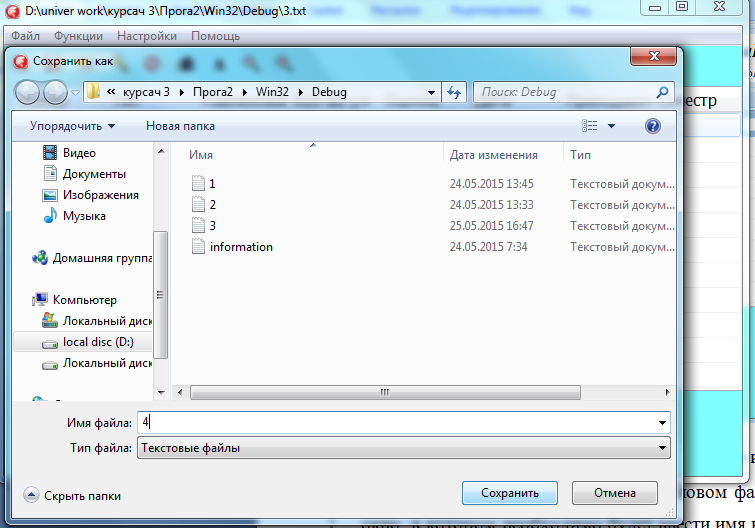


Рисунок 7 - Сохранение информации в новый файл

Для открытия сохраненного файла пользователю необходимо выбрать пункт меню «Открыть»:

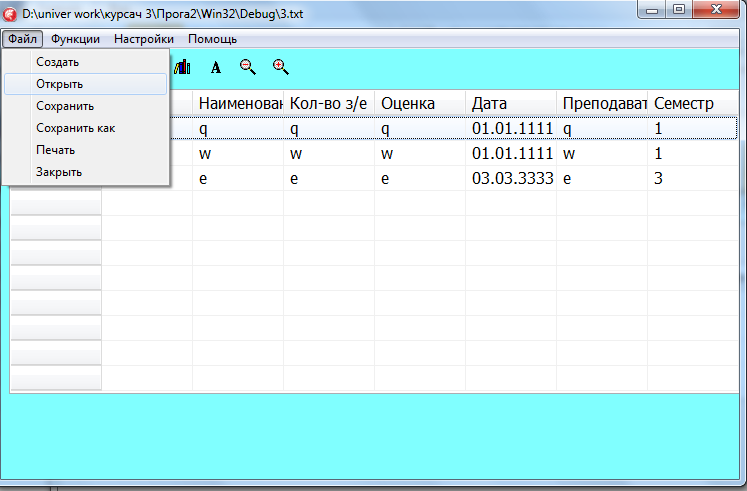


Рисунок 8 – Пункт меню для открытия файла

После этого необходимо найти требуемый файл и нажать кнопку «Открыть» в диалоговом окне:

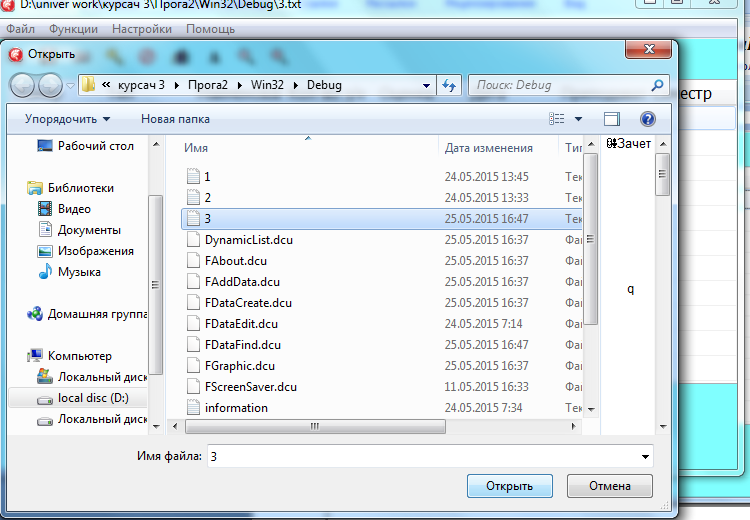


Рисунок 9 - Открытие файла

Для редактирования необходимо выбрать нужную запись, выбрать пункт меню «Редактировать», ввести новые значения и выбрать тип зачетной единицы.

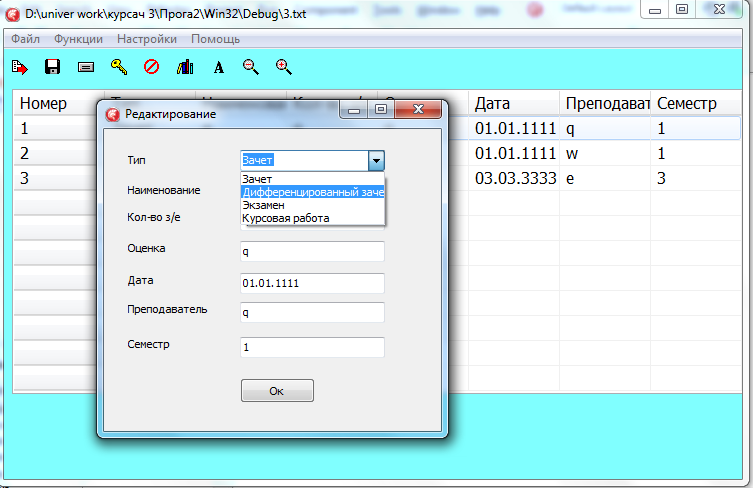


Рисунок 10 - Редактирование информации

Для поиска необходимо выбрать соответствующий пункт меню или нажать на кнопку на панели инструментов. Откроется окно поиска, после чего необходимо выбрать интересующий нас тип зачетной единицы и нажать кнопку «Ок».

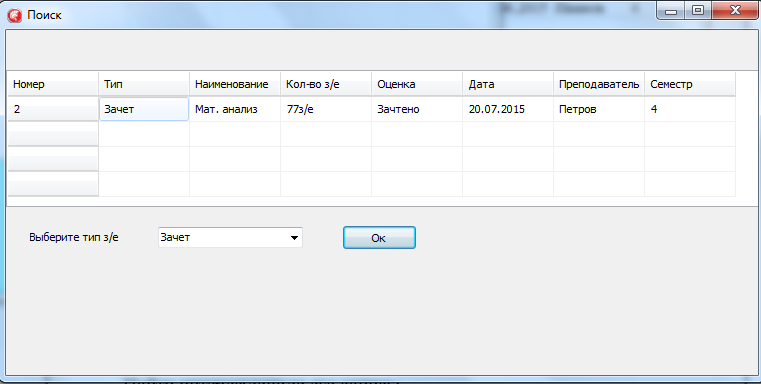


Рисунок 11 - Форма поиска

В случае, если ни одна запись не соответствует критериям поиска, то появится сообщение:

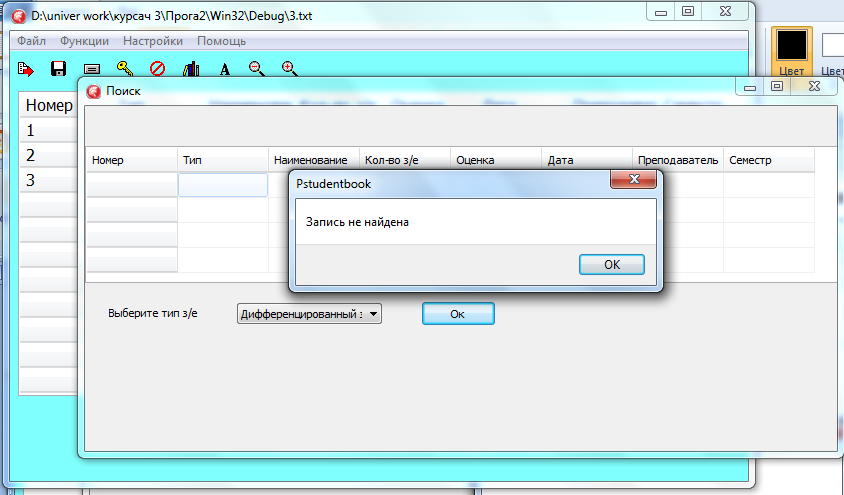


Рисунок 12 - Сообщение об отрицательном результате поиска

После чего пользователь может закрыть сообщение и продолжить работать с программой.

Для построения диаграммы необходимо выбрать пункт меню «Построить диаграмму» или нажать соответствующую кнопку на панели инструментов.

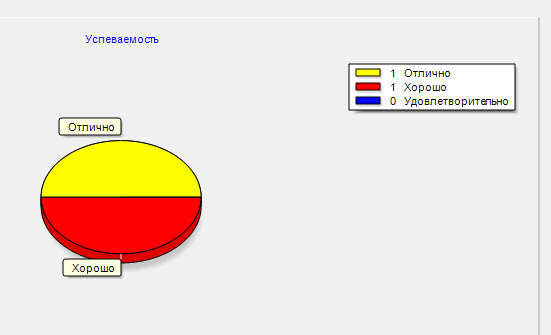


Рисунок 13 - Диаграмма

Заключение

В результате выполнения курсовой работы была разработана простая, удобная информационная система в IDE Embarcadero.

Были решены следующие задачи:

1. проведено исследование предметной области;
2. составлена информационная модель;
3. реализован графический интерфейс;
4. разработана функциональная часть программы;
5. отлажена работа программы.

Цель курсовой работы достигнута.

Список литературы

1. Акулов О. А. Информатика: базовый курс : учебник для студентов вузов, бакалавров, магистров, обучающихся по направлениям 552800, 654600 " Информатика и вычислительная техника" / О. А. Акулов, Н. В. Медведев.- М., 2010.– 557 с.
2. Архангельский А. Я. Программирование в Delphi : учебник по классическим версиям Delphi : [версии 5-7] / А. Я. Архангельский. - М., 2011.– 804 с.
3. Кобылянский В. Г. Хранение данных в ЭВМ : метод. указания к практ. занятиям и лаб. работам / В. Г. Кобылянский, Н. И. Агуленко ; ред. В. И. Хабаров ; Сиб. гос. ун-т путей сообщ. – Новосибирск : СГУПС, 2010.– 54 с.
4. Информатика и программирование : расчетно-графическое задание : методические указания для 1 курса ФБ специальности 080800 дневной и заочной формы обучения / Новосиб. гос. техн. ун-т ; [сост. М. Г. Зайцев ]. - Новосибирск, 2013. - 40 с.
5. Фуфаев, Э.В. Базы данных: Учебное пособие для студентов учреждений среднего профессионального образования / Э.В. Фуфаев, Д.Э. Фуфаев. - М.: ИЦ Академия, 2012. - 320 c.