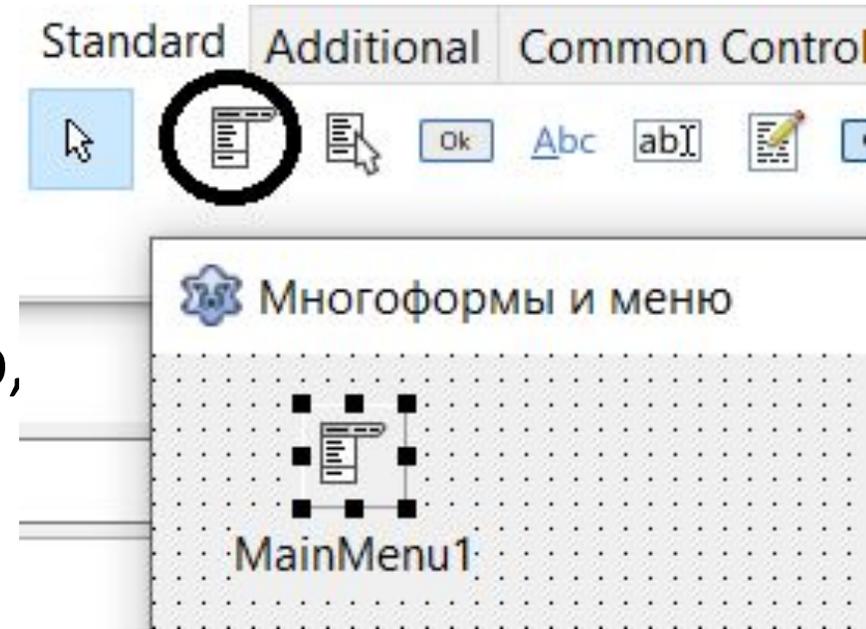


Обработка таблиц

Меню, многоформный интерфейс

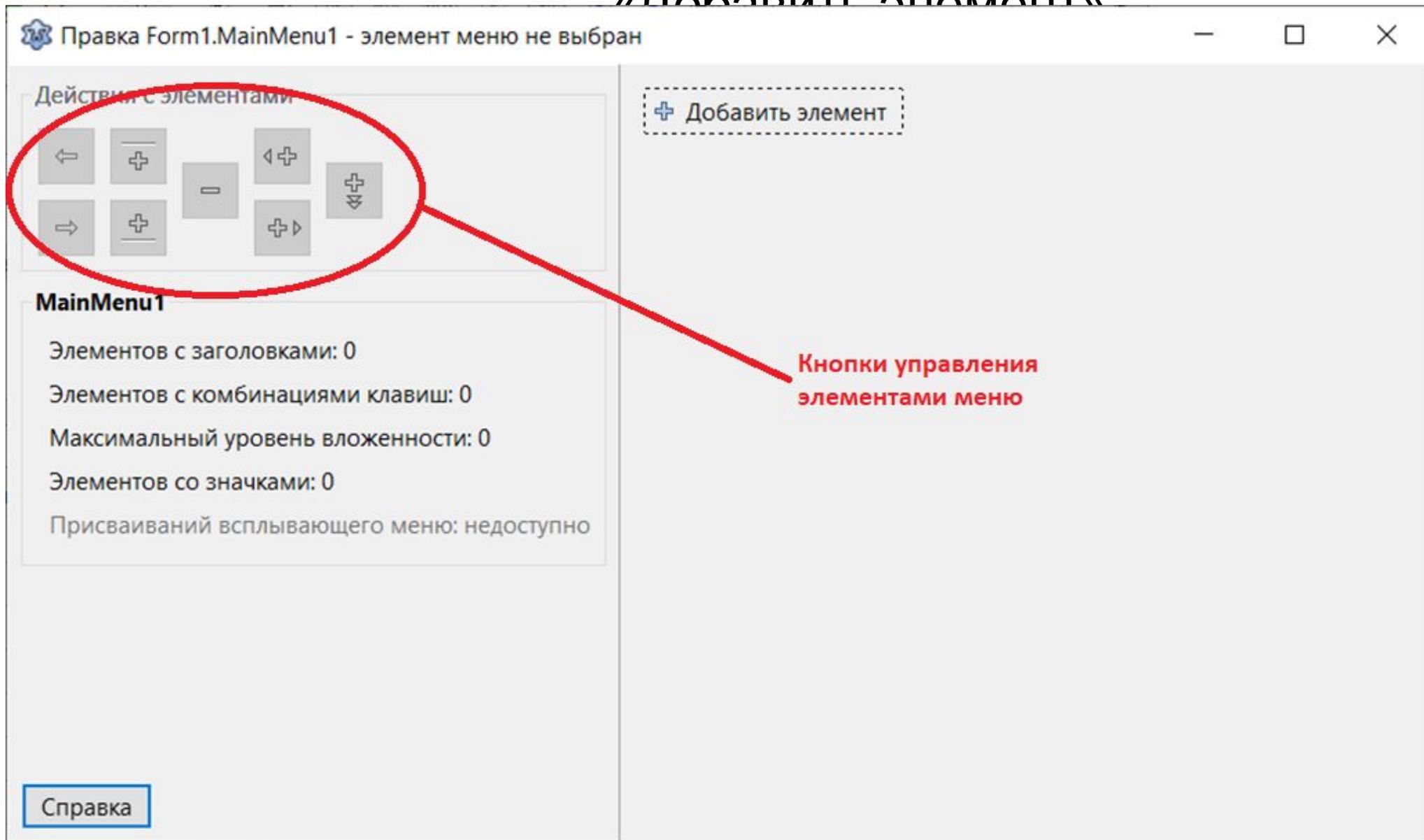
Компонента TMainMenu

- Предназначен для добавления к программе главного меню
- Чтобы добавить к приложению главное меню, нужно разместить на Форме в произвольном месте компонент TMainMenu.
- Компонент TMainMenu не визуальный, отображается на прототипе Формы как небольшой квадрат, в работающей программе не виден.
- Опции главного меню создаются с помощью специального редактора. Редактор меню вызывается с помощью двойного щелчка по компоненту TMainMenu.



Редактор Меню

Чтобы создать элемент меню, необходимо щелкнуть по кнопке «Добавить элемент»



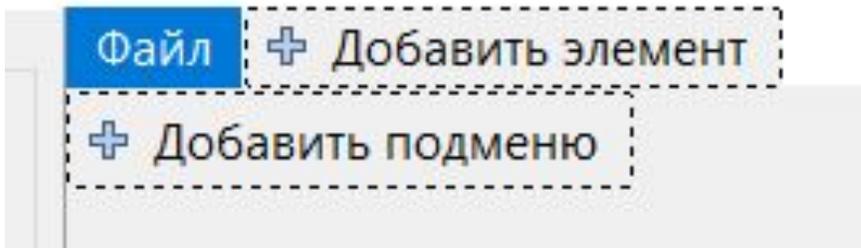
При этом появляется элемент с дефолтным именем MenuItem1. Для изменения заголовка надо нажать на правую кнопку мыши вызвав контекстное меню и выбрать пункт «Редактировать заголовок»

The screenshot shows the Visual Studio IDE interface. The title bar of the active window reads "Form1.MainMenu1.MenuItem1 - обработчик OnClick: не присвоен". The main area displays a context menu for the selected menu item "MenuItem1". The menu items are:

- Добавить элемент
- Переместить элемент налево (Ctrl+Вверх)
- Переместить элемент направо (Ctrl+Вниз)
- Разделители
- Удалить элемент (Del)
- Добавить новый элемент перед (Ins)
- Добавить новый элемент после
- Добавить подменю ниже (Ctrl+Ins)
- Добавить значок из ImageList
- Добавить обработчик OnClick
- Редактировать заголовок (Ввод)** (highlighted)
- Комбинации клавиш
- Шаблоны

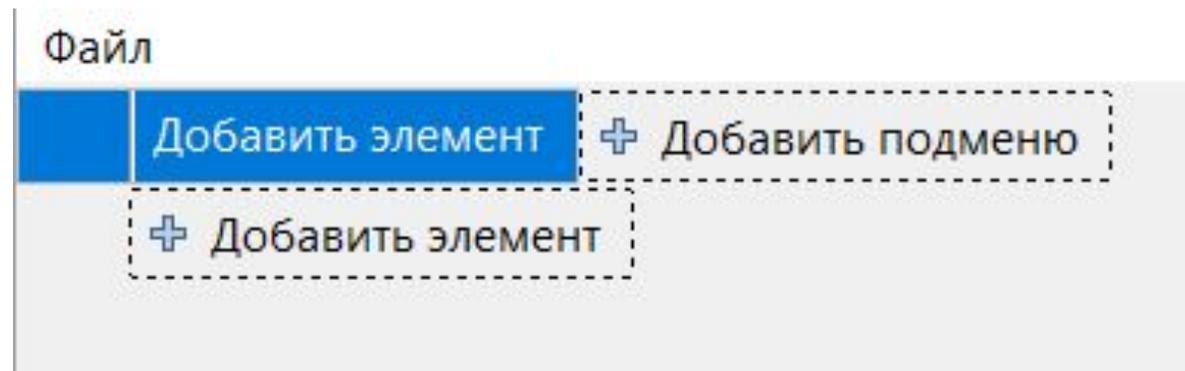
On the left side, the "Действия с элементами" (Actions) pane shows various icons for moving and deleting items. Below it, the "MainMenu1" pane displays the following statistics:

- Элементов с заголовками: 1
- Элементов с комбинациями клавиш: 0
- Максимальный уровень вложенности: 0
- Элементов со значками: 0
- Присваиваний всплывающего меню: недоступно

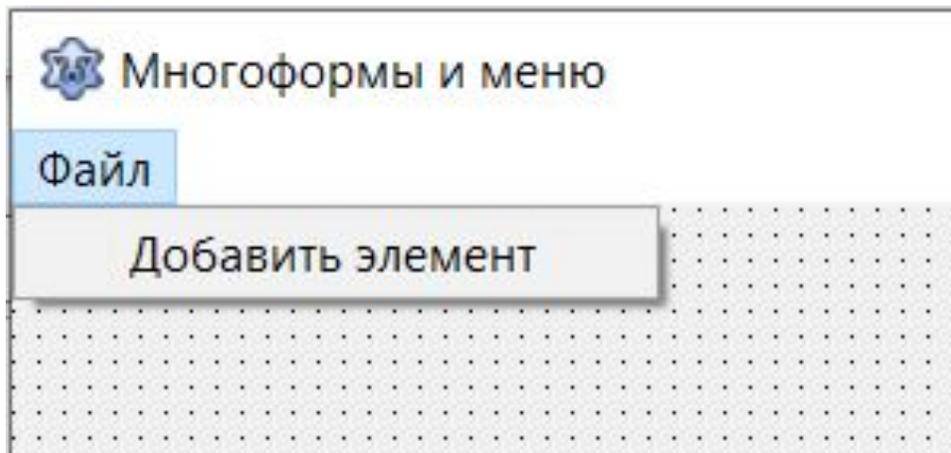


Чтобы добавить пункт выпадающего меню, необходимо щелкнуть по кнопке ниже

Далее возможно добавлять дополнительные элементы меню



Для обработки события выбора меню, необходимо дважды щелкнуть на пункте, при этом автоматически создается



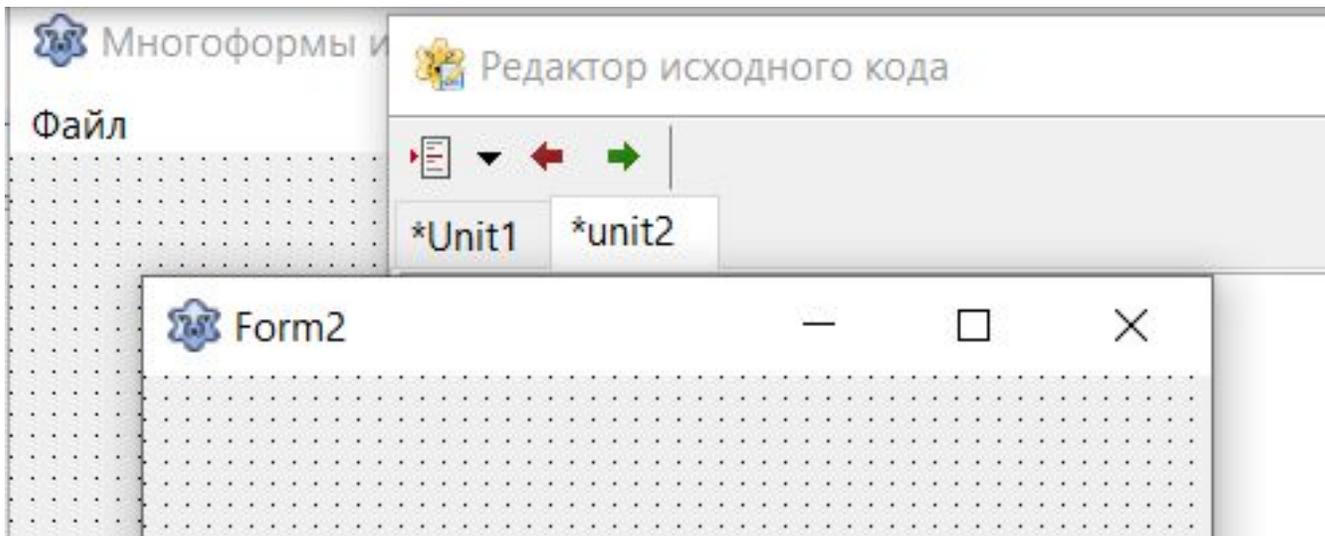
<- Результат

Многоформный однодокументный интерфейс

Для добавления новой формы к проекту используем последовательность меню: Файл -> Создать Форму

При этом автоматически создается новая форма, в редакторе исходного кода появляется новая вкладка Unit2 для описания событий новой формы

```
uses  
Classes, SysUtils, Forms, Controls, Graphics, Dialogs, Menus, Unit2;
```



Открыть дополнительную форму можно по наступлению некоторого события на 1 форме.

Пусть это событие будет – выбор пункта меню «Добавить элемент».

Дважды кликаем по нужному пункту меню, создается процедура:

```
procedure TForm1.MenuItem2Click(Sender: TObject);
```

```
begin
```

```
  Form2.Show;
```

```
end;
```

Метод **Show** открывает дополнительную форму. При этом обе формы работают одновременно.

Также для открытия дополнительной формы можно использовать метод **ShowModal** в этом случае управление полностью переходит к дочернему окну. Выполнение события первой формы приостанавливается до закрытия дочерней.

Form2.Close – закрыть форму 2

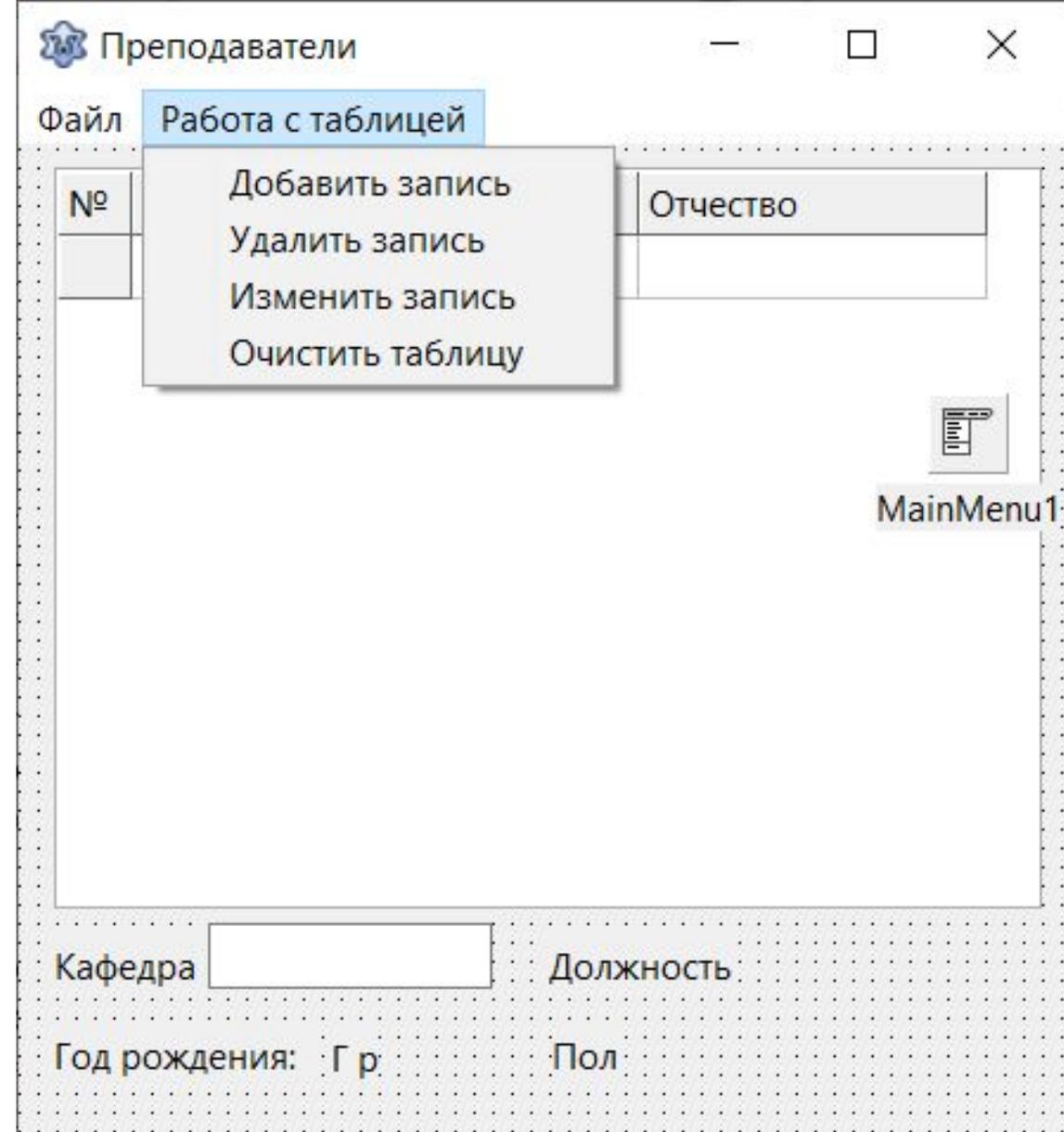
Пример работы с доп. формой и меню

Основная форма содержит компонент StringGrid – таблица отражающая список преподавателей.

Таблица имеет следующие столбцы: Номер по порядку, Фамилия, Имя, Отчество преподавателя.

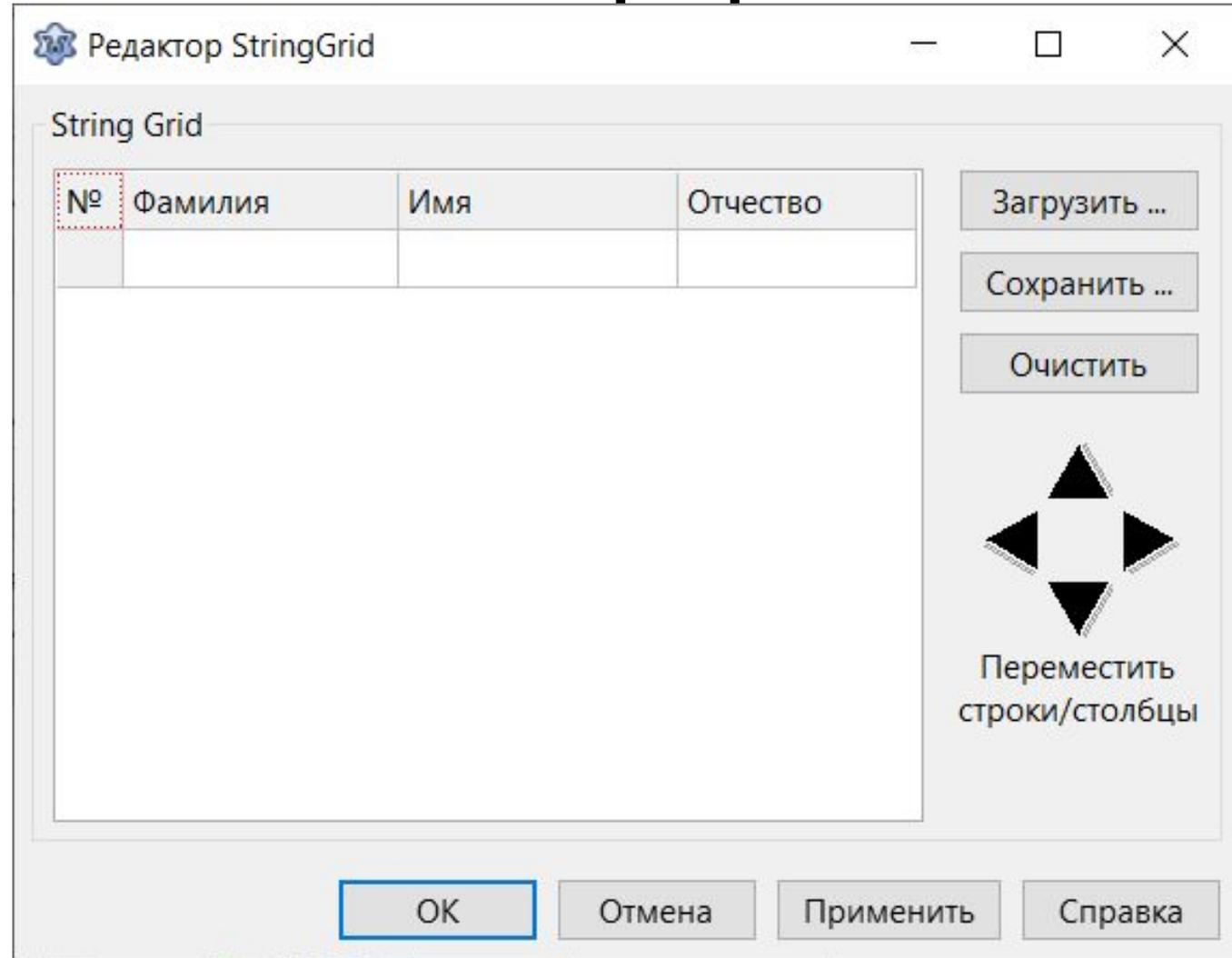
В компоненте Edit отображается кафедра выделенного преподавателя

Также на форме присутствуют Label, где отображаются должность человека, пол, год рождения



Установки StringGrid на основной форме

1. Установить компонент StringGrid на форму.
2. ColCount = 4; RowCount = 2;
3. FixedCol = 1; FixedRow = 1;
4. Дважды кликаем на компоненте -> попадаем в редактор StringGrid
5. Устанавливаем нужную ширину колонок
6. Устанавливаем необходимые заголовки
7. В опциях goEditing = False



Проектирование дополнительной формы – для ввода одного элемента таблицы

- На форме 5 элементов Edit для ввода соответствующей информации
- 1 элемент ComboBox – для выбора должности
- 6 элементов Label – заголовки для пояснения ввода
- 1 элемент RadioGroup для выбора пола
- 2 кнопки:
 - ✓ ОК – если все в порядке, и элемент можно поместить в таблицу
 - ✓ Отмена – если пользователь передумал и таблица не изменяется.

Ввод элем...

Фамилия:

Имя:

Отчество:

Кафедра:

Должность:

Год рождения:

Пол

Муж Жен

Отмена ОК

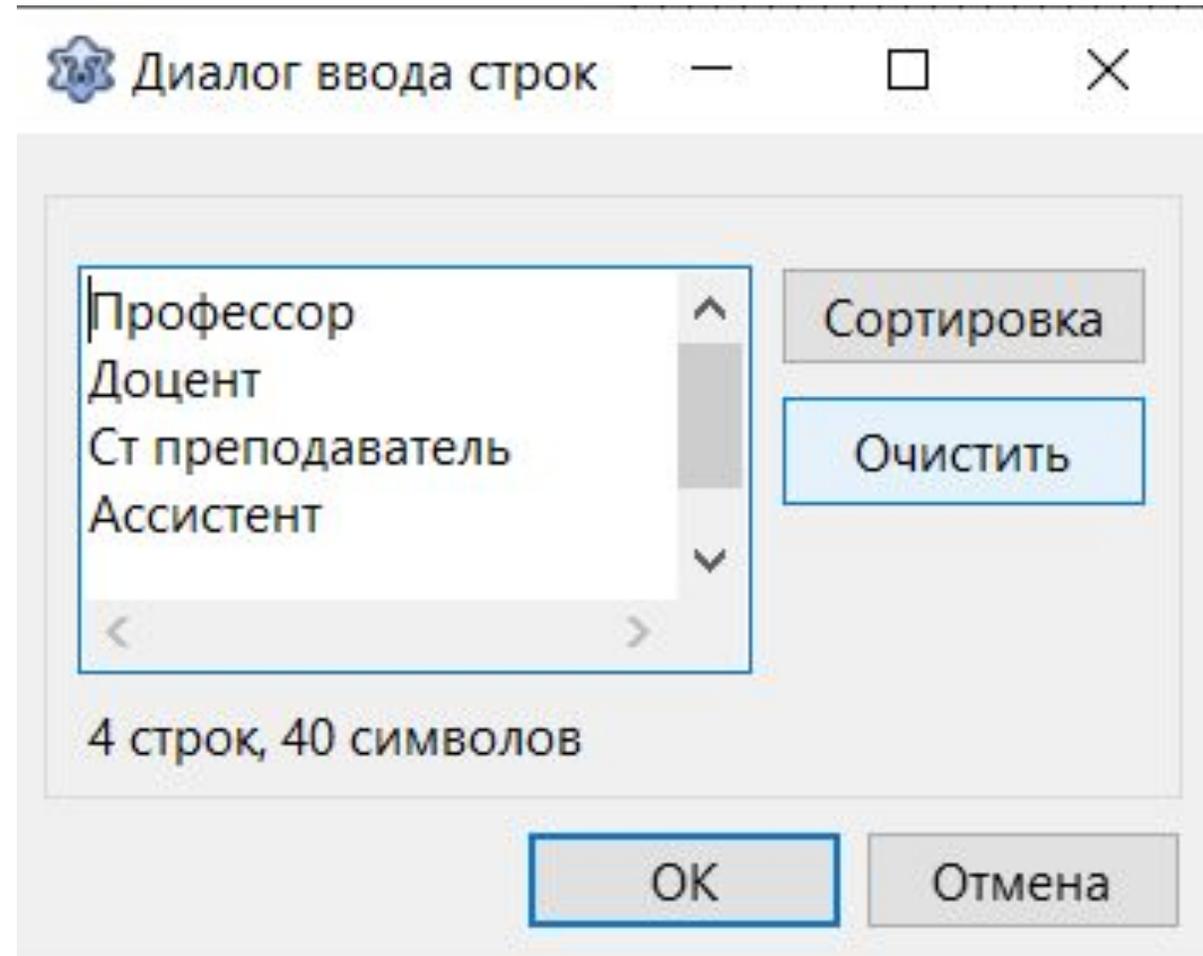
В компоненте ComboBox необходимо установить возможные варианты для выбора.

Для используется редактор свойства Items. Чтобы его открыть, надо щелкнуть по кнопке «...». Появится диалог ввода строк.

В редакторе можно ввести должности

Свойство Text компоненте должно быть пустым.

Для элемента Edit5 (ввод года рождения) свойство NumbersOnly необходимо переключить в значение True.



Работа с RadioGroup

Компонент предназначен для выбора одного элемента из предложенных.

Находится в панели инструментов в закладке «Standart»

Свойству Caption назначаем «Пол»

Активизируем свойство Items, в диалоговом окне пишем 2 строки:

- Муж
- Жен

Свойству Columns назначаем 2. То есть у нас строки будут распределены на 2 колонки.

Свойству ItemIndex назначаем 0 . Это свойство определяет номер строки, которая в данный момент выбрана. Значение -1 говорит о том, что не выбрана ни одна строка.

Глобальные типы и переменные

Описываются в разделе **interface** дочернего модуля

В блоке **type** добавляется новый тип запись и новая таблица типа запись:

```
prep = Record
```

```
    fam,name,otch: string[50];
```

```
    kaf,dol:string[40];
```

```
    gen:byte;
```

```
    brn:word;
```

```
end;
```

```
tab = array [1..50] of prep;
```

В блоке **var** добавляются переменные:

```
t:tab; n:byte; flag:boolean //флаг для проверки возврата (ок или отмена)
```

Нумерация элементов таблицы начинается с 1. В этом случае номер элемента таблицы будет совпадать с номером в StringGrid, так как нулевая строка компонента в используется для заголовков.

Программирование доп. формы

Для дополнительного модуля необходимо написать события нажатия на кнопки

```
procedure TForm2.Button2Click(Sender: TObject); //Кнопка Отмена
```

```
begin
```

```
  flag:=False; //передается запрет на изменения осн. таблицы
```

```
  Form2.Close;
```

```
end;
```

```
procedure TForm2.Button1Click(Sender: TObject); //Кнопка ОК
```

```
begin
```

```
  flag:=True; //передается разрешение на изменения осн. таблицы
```

```
  Form2.Close;
```

```
end;
```

Программирование основной формы

Компонента StringGrid служит для отображения таблицы записей. Поэтому целесообразно написать отдельно такую процедуру. Назовем ее UpdateTab.

Это будет процедура класса TForm, поэтому описать ее нужно в классе

```
TForm1 = class (TForm)
    Label2: TLabel;
    Label3: TLabel;
    MainMenu1: TMainMenu;
    MenuItem1: TMenuItem;
    MenuItem3: TMenuItem;
    MenuItem4: TMenuItem;
    MenuItem5: TMenuItem;
    MenuItem6: TMenuItem;
    MenuItem7: TMenuItem;
    StringGrid1: TStringGrid;
    procedure MenuItem4Click(Sender: TObject);
    procedure UpdateTab();
private
public
```

В разделе implementation напишем тело процедуры

```
procedure TForm1.UpdateTab();
```

```
var i:byte;
```

```
begin
```

```
  StringGrid1.RowCount:=n+1;      //количество элементов + заголовки
```

```
  for i:=1 to n do begin
```

```
    StringGrid1.cells[0,i]:=IntToStr(i);    //номер в таблице
```

```
    StringGrid1.cells[1,i]:=t[i].fam;
```

```
    StringGrid1.cells[2,i]:=t[i].name;
```

```
    StringGrid1.cells[3,i]:=t[i].otch;
```

```
  end;
```

```
  StringGrid1.Row:=1;      //По умолчанию активизируем первую строку
```

```
  WhenSelecting(1);      //Обращаемся к процедуре заполнения
```

```
end;          //других элементов формы (описана ниже)
```

При активизации строки таблицы нам надо, чтобы в поле Edit и в label отразились соответствующие текущему элементу данные. Для этого также напишем общую процедуру, которая это делает. Назовем ее ***WhenSelecting(i:integer)***. Параметр *i* передает номер активной строки. Описание процедуры также необходимо выполнить в Классе TForm.

```
procedure TForm1.WhenSelecting(i:integer);
```

```
Begin
```

```
Edit1.Text:=t[i].kaf;
```

```
Label3.Caption:=t[i].dol;
```

```
Label4.Caption:=IntToStr(t[i].brn);
```

```
if t[i].gen=0 then Label5.Caption:='Мужчина' else  
Label5.Caption:='Женщина';
```

```
end;
```

Обработка события «Выбор пункта меню «Добавить запись»»

Прежде чем кодить, надо понять, что происходит при наступлении события.

В данном случае необходимо, чтобы открылась дополнительная форма для ввода информации. После того, как данные были заполнены, они попадают на основную форму, в таблицу.

Следовательно, дополнительную форму надо открывать в модальном режиме. Для того, чтобы на основной форме прервать выполнение программы. Она продолжится после закрытия второй формы, когда на ней будут данные.

Меню – Добавить запись

```
procedure TForm1.MenuItem4Click(Sender: TObject);
```

```
begin
```

```
Form2.Edit1.Text:=""; Form2.Edit2.Text:="";
```

```
Form2.Edit3.Text:=""; Form2.Edit4.Text:="";
```

```
Form2.Edit5.Text:="";
```

```
Form2.RadioGroup1.ItemIndex:=0;
```

```
Form2.ComboBox1.Text:="";
```

```
Form2.ShowModal;
```

```
if flag then begin
```

```
  n:=n+1;
```

```
  t[n].fam:=Form2.Edit1.Text;
```

```
  t[n].name:=Form2.Edit2.Text;
```

```
  t[n].otch:=Form2.Edit3.Text;
```

```
  t[n].kaf:=Form2.Edit4.Text;
```

```
  t[n].brn:=StrToInt(Form2.Edit5.Text);
```

```
  t[n].dol:=Form2.ComboBox1.Text;
```

```
  t[n].gen:=Form2.RadioGroup1.ItemIndex; end;
```

```
UpdateTab();
```

```
end;
```

Очищаем поля ввода на 2 форме

Модально открываем 2 окно

Выполняется только в случае возврата флага TRUE. И только после закрытия второй формы.

Количество элементов таблицы увеличивается на 1, данные полей ввода дополнительной формы записываем в новый элемент таблицы

Обновляем таблицу в StringGrid

Обработка события – выбор меню «Изменить запись»

Идея в том, чтобы поправить неверно введенные записи. Не целиком удалить и затем ввести новую, а только поправить.

В этом случае, при выборе пункта меню, открывается то же окно, но уже не с пустыми полями, а со значениями полей текущего элемента.

Дополнительная форма также вызывается в модальном режиме.

После закрытия, данные перезаписываются в текущий элемент таблицы и отображаются на 1 форме.

Меню – Изменить запись

```
procedure TForm1.MenuItem6Click(Sender: TObject);
```

```
var k:byte;
```

```
begin
```

```
  k:=StringGrid1.Row;
```

Определяем номер текущей строки. Он же будет номером элемента в таблице

```
  Form2.Edit1.Text:=t[k].fam;    Form2.Edit2.Text:=t[k].name;
```

```
  Form2.Edit3.Text:=t[k].otch;   Form2.Edit4.Text:=t[k].kaf;
```

```
  Form2.Edit5.Text:=IntToStr(t[k].brn);
```

```
  Form2.RadioGroup1.ItemIndex:=t[k].gen;
```

```
  Form2.ComboBox1.Text:=t[k].dol;
```

В поля ввода 2 формы помещаем данные k-ого элемента

```
  Form2.ShowModal;
```

Модально открываем 2 окно

```
  if flag then begin
```

```
    t[k].fam:=Form2.Edit1.Text;  t[k].name:=Form2.Edit2.Text;
```

```
    t[k].otch:=Form2.Edit3.Text; t[k].kaf:=Form2.Edit4.Text;
```

```
    t[k].brn:=StrToInt(Form2.Edit5.Text);
```

```
    t[k].dol:=Form2.ComboBox1.Text;
```

```
    t[k].gen:=Form2.RadioGroup1.ItemIndex; end;
```

Выполняется только в случае возврата флага TRUE. И только после закрытия второй формы. данные полей ввода дополнительной формы записываем в элемент таблицы номер k

```
  UpdateTab();
```

Обновляем таблицу в StringGrid

```
end;
```

Меню «Удалить запись»

При наступлении события необходимо удалить текущую запись:

```
procedure TForm1.MenuItem5Click(Sender: TObject);
var k,j:byte;
begin
  k:=StringGrid1.Row; //определяем номер текущей строки = номер записи
  for j:=k to n-1 do //Все элементы после текущей сдвигаем на 1 вперед
    t[j]:=t[j+1];
  n:=n-1; //Уменьшаем количество элементов таблицы на 1
  UpdateTab(); //Обновляем вывод таблицы
```

Меню «Очистить таблицу»

```
procedure TForm1.MenuItem7Click(Sender: TObject);
```

```
begin
```

```
  n:=0;           //обнулить количество элементов
```

```
  StringGrid1.RowCount:=2;      //в StringGrid оставить заголовки и 1 строку
```

```
  StringGrid1.Rows[1].Clear;    //Очистить эту пустую строку
```

```
  Edit1.Text:="";             //Очистить поле ввода Edit
```

```
  Label3.Caption:='Должность';  //Метки тоже очищаются
```

```
  Label4.Caption:='--';
```

```
  Label5.Caption:='Пол';
```

```
end;
```

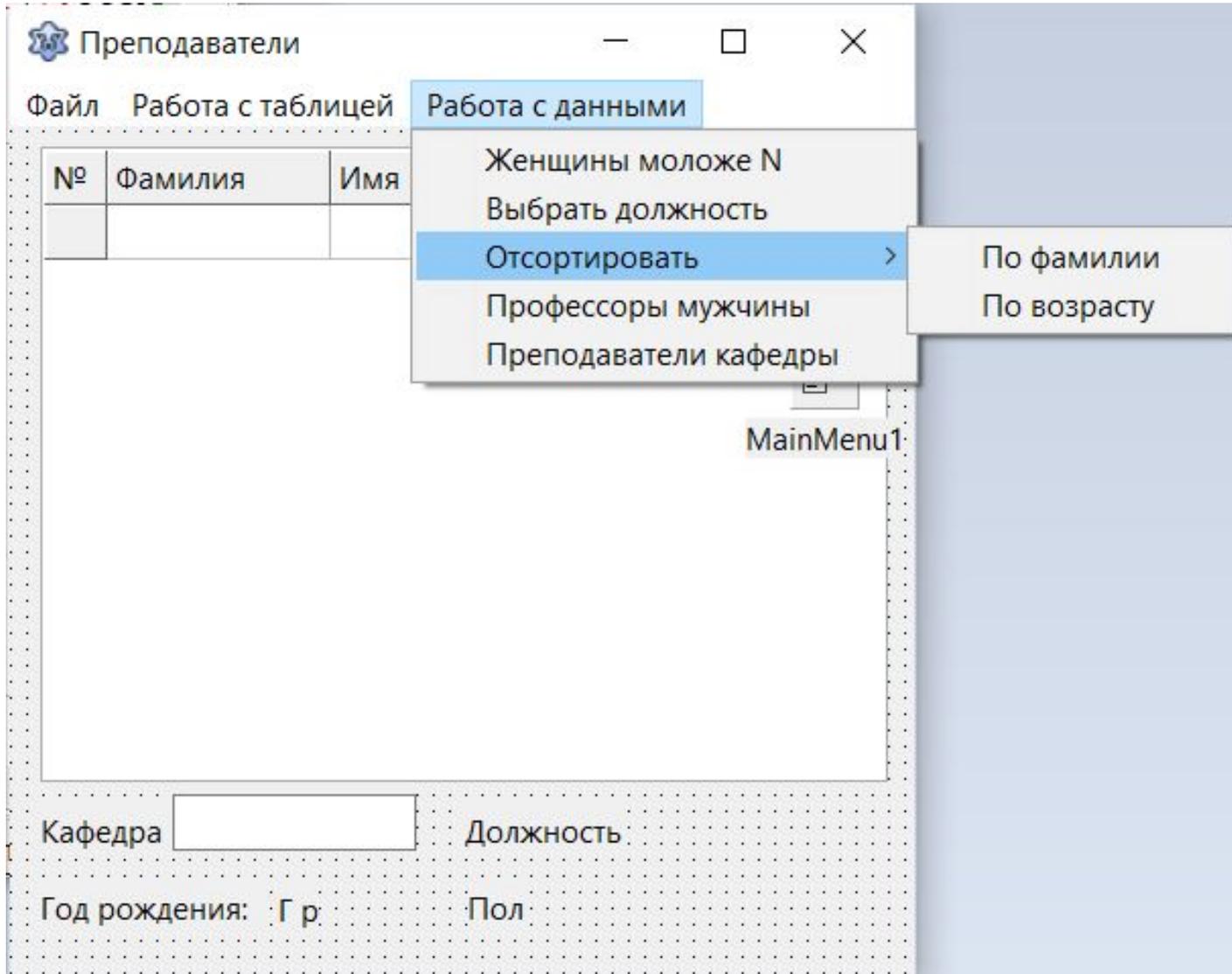
Обработка данных таблицы

Данные таблицы можно сортировать, отбирать, компоновать.

1. Выбрать всех преподавателей женщин моложе N лет
2. Выбрать всех преподавателей заданной должности
3. Отсортировать преподавателей по фамилии, по возрасту
4. Выбрать всех профессоров мужчин
5. Выбрать всех преподавателей заданной кафедры

Результаты (кроме сортировок) выводить в отдельной форме в компонент Метод

Доработка МЕНЮ



Добавляем еще несколько пунктов меню. Для этого:

- Дважды щелкаем на иконку формы MainMenu1
- Щелкаем на последний пункт основного меню “Работа с таблицей”
- Добавляем элементы в необходимом нам месте.

<- результат

Дополнительная форма для вывода

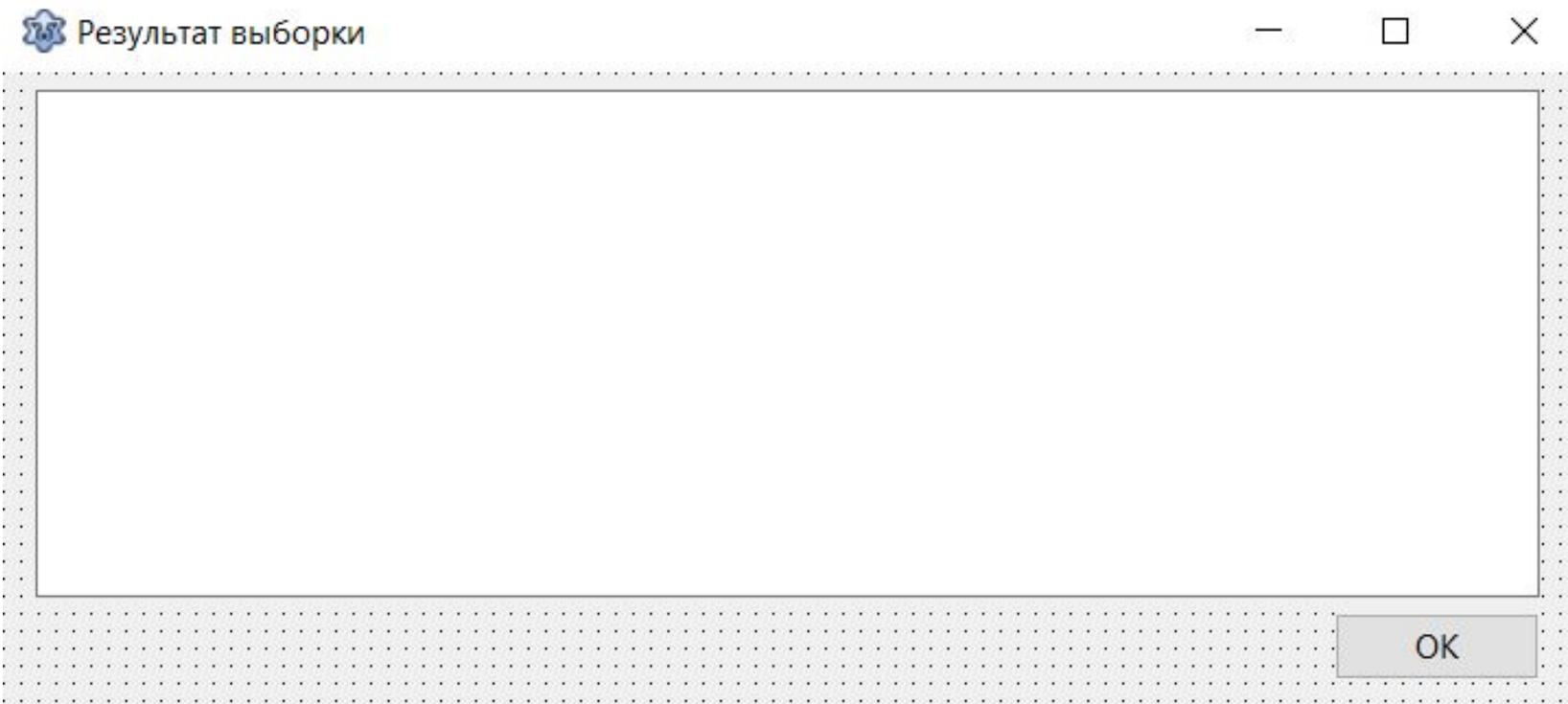
результатов

Создание новой формы: Файл -> Создать форму

На форму помещаем компонент Memo

Свойство ReadOnly устанавливаем в положение True

Заголовок – «Результат выборки»



Добавляем кнопку для закрытия формы с заголовком «ОК».

Подсоединяем Unit3 к Unit1

Выбрать всех преподавателей женщин моложе N

Создать обработку выполнения п. меню можно двойным щелчком на этом пункте.

```
procedure TForm1.MenuItem10Click(Sender: TObject);
var st:string; cd,j, v:byte;
begin
  repeat
    st:=InputBox('Ввод данных','Укажите возраст,");
    val(st,v,cd);
  until cd=0;
  Form3.Memo1.Clear;
  Form3.Memo1.Lines.Add('Женщины моложе '+IntToStr(v));
  for j:=1 to n do
    if ((2020 - t[j].brn) < v) and (t[j].gen=1) Then
      Form3.Memo1.Lines.Add(t[j].fam+' '+t[j].name+' '+t[j].otch+' '+t[j].dol
        +' '+t[j].kaf+' '+IntToStr(t[j].brn));
  Form3.Show;
end;
```

Выбрать всех преподавателей заданной ДОЛЖНОСТИ

```
procedure TForm1.MenuItem1Click(Sender: TObject);
var st:string[30]; j:byte;
begin
  st:=InputBox('Ввод данных','Введите должность,');
  Form3.Memo1.Clear;
  Form3.Memo1.Lines.Add('Преподаватели с должностью '+st);
  for j:=1 to n do
    if t[j].dol=st Then
      Form3.Memo1.Lines.Add(t[j].fam+' '+t[j].name+' '+t[j].otch+' '+t[j].dol
        +' '+t[j].kaf+' '+IntToStr(t[j].brn));
  Form3.show;
end;
```

Отсортировать преподавателей по фамилии

```
procedure TForm1.MenuItem13Click(Sender: TObject);
var e,x:byte; buf:prep;
begin
  for e:=1 to n-1 do
    for x:=1 to n-1 do
      if t[x].fam > t[x+1].fam then begin
        buf:=t[x];
        t[x]:=t[x+1];
        t[x+1]:=buf;
      end;
    UpdateTab();
  end;
```

Отсортировать преподавателей по возрасту

```
procedure TForm1.MenuItem14Click(Sender: TObject);
var e,x:byte; buf:prep;
begin
  for e:=1 to n-1 do
    for x:=1 to n-1 do
      if t[x].brn < t[x+1].brn then begin
        buf:=t[x];
        t[x]:=t[x+1];
        t[x+1]:=buf;
      end;
    UpdateTab();
  end;
```

Выбрать всех профессоров

МУЖЧИН

```
procedure TForm1.MenuItem15Click(Sender: TObject);
var j:byte;
begin
  Form3.Memo1.Clear;
  Form3.Memo1.Lines.Add('Мужчины профессоры ');
  for j:=1 to n do
    if (t[j].dol='Профессор') and (t[j].gen=0) Then
      Form3.Memo1.Lines.Add(t[j].fam+' '+t[j].name+' '+t[j].otch+' '+t[j].dol
        +' '+t[j].kaf+' '+IntToStr(t[j].brn));
  Form3.Show;
end;
```

Выбрать всех преподавателей заданной кафедры

```
procedure TForm1.MenuItem16Click(Sender: TObject);
var st:string[30]; j:byte;
begin
  st:=InputBox('Ввод данных','Введите кафедру','');
  Form3.Memo1.Clear;
  Form3.Memo1.Lines.Add('Преподаватели кафедры '+st);
  for j:=1 to n do
    if t[j].kaf=st Then
      Form3.Memo1.Lines.Add(t[j].fam+' '+t[j].name+' '+t[j].otch+' '+t[j].dol
        +' '+t[j].kaf+' '+IntToStr(t[j].brn));
  Form3.show;
end;
```

Задание на лабораторную работу

1. Создать таблицу с полями: Фамилия студента, индекс группы, три оценки. Вывести данные в компонент `StringGrid`
2. Создать меню, обеспечивающее ввод данных в таблицу, обработку данных таблицы, очистку таблицы и закрытие приложения.
3. Ввод новых элементов обеспечить с использованием дополнительной формы
4. Вывод по запросу:
 - номер группы с наивысшим средним баллом;
 - фамилии студентов – отличников (неуспевающих);
 - средний балл студента по его фамилии.
5. Упорядочить студентов по фамилии, по возрастанию среднего балла

Способ вывода по номеру студента