

## Лабораторная работа №30,31

**Тема.** Объявление класса, создание экземпляров класса. Создание наследованного класса.

**Задание.** Составить программу, которая формирует и выводит список студентов и преподавателей. В ходе разработки приложения использовать концепции наследования и полиморфизма.

### Ход работы

1. Откройте новое приложение Delphi и на форме разместите компоненты так, как показано на рисунке 1

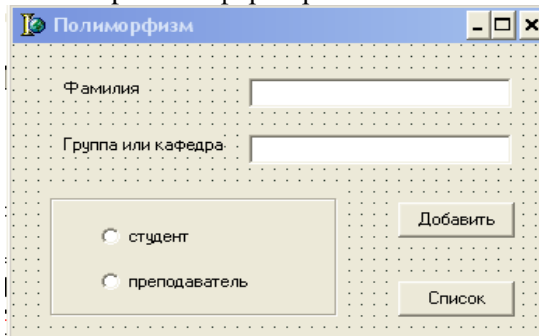


Рисунок 1. Диалоговое окно программы Полиморфизм

2. Создайте код программы, используя листинг приложения, приведенный ниже:

```
unit Unit1;
interface
uses Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs, StdCtrls, ExtCtrls;
Type TForm1 = class(TForm)
  Label1: TLabel;
  Label2: TLabel;
  Edit1: TEdit;
  Edit2: TEdit;
  Button1: TButton;
  Button2: TButton;
  GroupBox1: TGroupBox;
  RadioButton1: TRadioButton;
  RadioButton2: TRadioButton;
  procedure Button1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
private { Private declarations }
public { Public declarations }
end;
Type
//базовый класс
TPerson = class //класс
  fname : string; //имя
  constructor Create(Name : string);
  function Info : string; virtual;
end;

//класс Студент
TStud = class(TPerson)
  fgr : integer; //номер группы
  constructor Create(Name : string; gr : integer);
  function Info : string; override;
end;

//класс Преподаватель
TProf = class(TPerson)
  fdep : string; //название кафедры
  constructor Create(Name : string; dep : string);
  function Info : string; override;
end;
const SZL = 10; //размер списка
var Form1: TForm1;
  List : array[1..SZL] of TPerson; //список
  n : integer = 0; //количество людей в списке
implementation
{$R *.dfm}
constructor TPerson.Create(name : string);
```

```

begin fName := name;
end;

constructor TStud.Create(name : string; gr : integer);
begin
  inherited create(name); //вызвать конструктор базового класса
  fGr := gr;
end;

constructor TProf.Create(name : string; dep : string);
begin
  inherited create(name); //вызвать конструктор базового класса
  fDep := dep;
end;

function TPerson.Info : string;
begin
  result := fName;
end;

function TStud.Info : string;
begin
  result := fName + ' гр. ' + IntToStr(fgr);
end;

function TProf.Info : string;
begin
  result := fName + ' каф. ' + fdep;
end;

//щелчок на кнопке Добавить
procedure TForm1.Button1Click(Sender: TObject);
begin
  if n < SZL then begin
    //добавить объект в список
    inc(n);
    if RadioButton1.Checked
    then List[n]:=TStud.Create(Edit1.Text, StrToInt(Edit2.Text)) //создание объекта TStud
    else //создание объекта TProf
      List[n]:=TProf.Create(Edit1.Text, Edit2.Text);
    //очистка полей ввода
    Edit1.Text := ''; Edit2.Text := '';
    Edit1.SetFocus; //курсор в поле Фамилия
  End else ShowMessage('Список заполнен');
end;

//щелчок на кнопке Список
procedure TForm1.Button2Click(Sender: TObject);
var i : integer; //индекс
    st : string; //список
begin
  for i := 1 to SZL do if list[i]<>NIL then st := st + list[i].Info + #13;
  ShowMessage('Список'+#13+st);
end;end.

```

**Отчет должен содержать:** название, цель и задание работы, листинг модулей приложения, ответы на контрольные вопросы.

### Контрольные вопросы

1. Дайте определение концепциям ООП: инкапсуляции, наследованию, полиморфизму.
2. Что такое делегирование?
3. Опишите назначение следующих директив: private, protected, virtual, override.
4. Какие существуют методы для установки значений свойства?
5. Что такое класс, событие, метод, поле, свойство, объект?