

Практическая работа №10, ТЕСТ ПО ГЕОГРАФИИ

Постановка задачи

Создайте проект, состоящий из нескольких форм: титульной формы, формы меню, формы помощи и формы непосредственно теста. В зависимости от правильности ответов на тест, который состоит из нескольких разделов (в свою очередь в разделе может быть несколько вопросов), подсчитывается результат и выставляется оценка по каждому разделу.

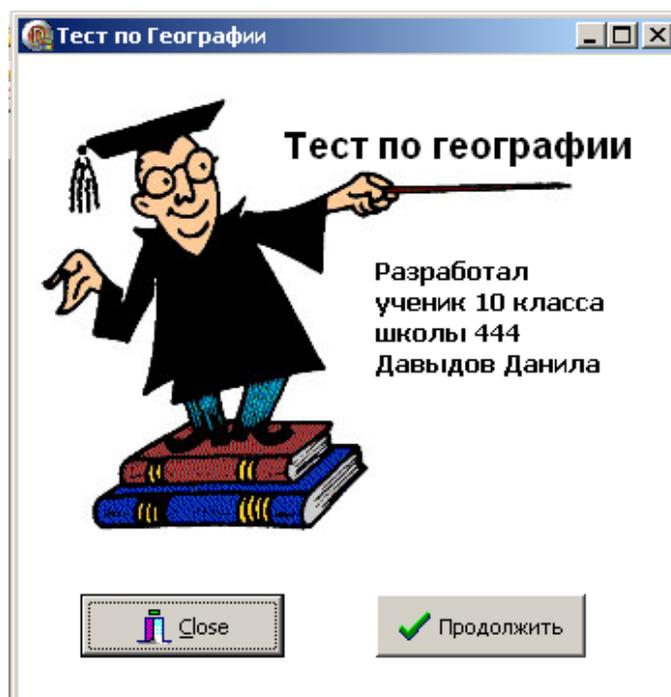


Рис.22

Новыми в этой программе являются:

- использование в проекте нескольких форм и организация вызова этих форм,
- использование шаблона формы **Tabbed pages** из окна **New Items** вкладки **Forms**,
- создание и использование компонента главное меню формы **MainMenu**, палитры компонентов **Standard**,
- компонента **Memo** (многострочный текстовый редактор),
- использование компонента **Image** палитры компонентов **Additional** для размещения графического объекта,
- компонента **PageControl** (набор панелей с закладками).

План разработки программы

1. Состав проекта

Откройте новый проект. Работа над проектом состоит из последовательного создания пяти форм:

- Титульная форма,
- Форма меню,
- Форма инструкции,

- Форма тестирования,
- Форма результата.

2. Титульная форма

2.1. Создайте титульную форму. Дайте название форме «Тест по Географии».

2.2. Разместите в форме компонент **Image** (графический объект), который находится на вкладке **Additional** палитры компонентов. Выделите **Image**, справа от свойства **Picture** щелкните на кнопке с тремя точками. В появившемся окне встроенного редактора **Picture Editor** щелкните кнопку «Load», которая вызывает открытие окна для установки имени графического файла. Найдите на диске нужный графический файл и щелкните на его имени. Выбранная картинка появится вначале в служебном окне, и если пользователь подтвердит (кнопка «ОК»), то картинка появится и в окне формы.

2.3. Изображение, которое разместили на форме, имеет белый фон. Для того, чтобы изображение резко не выделялось на форме поменяем цвет фона формы на белый – значение свойства **Color clWhite**.

2.4. Разместите в форме метки в соответствии с рис.22 и присвойте им заголовки.

2.5. Разместите в форме кнопку «Выход», обеспечивающую прекращение выполнения задачи.

2.6. Разместите в форме кнопку «Продолжить». Пока не задаем действия, которые будут выполняться по нажатию этой кнопки.

2.6. Сохраните форму под именем, например, **F_Titul.pas** и проект – **Pr10.dpr**.

3. Форма «Меню»

3.1. Создайте новую форму (U_Menu) с помощью команды **File/New/Form**. Выполните следующие действия.

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/Имя события	Значение/Действие
Form2	Properties	Caption	Меню
		Name	U_Menu установка имени формы, под которым компонент будет известен программе.

3.2. Создайте меню с помощью компонента **MainMenu**, который находится на странице палитры компонентов **Standard**. Теперь необходимо создать его опции (пункты меню).

Дважды щелкните на появившейся пиктограмме **MainMenu** или, выделив компонент **MainMenu**, дважды щелкните по кнопке в правой половине строки свойства **Items**. И в том и в другом случае на экране появится окно с заголовком **U_Menu.MainMenu1** (рис.23).

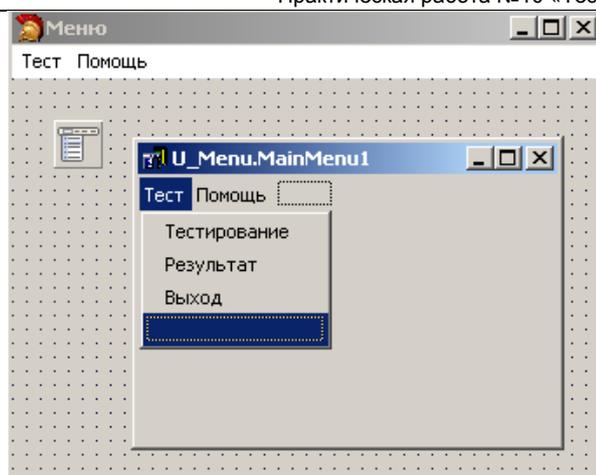


Рис.23

Установите в качестве значения свойства **Caption** название первого пункта меню «Тест». Оно запишется в синем прямоугольнике слева на полосе меню. Нажмите клавишу «Enter». Теперь опция готова и можно переходить к созданию следующей опции.

Каждая опция главного меню может содержать список подопций. Для создания их щелкните мышью по строке ниже опции и введите первую подопцию. Продолжайте ввод, пока не будет создан весь список подопций.

В названиях опций, являющихся значениями свойства **Caption**, можно указать символ «&» перед тем символом, который будет использоваться для быстрого выбора опции. В нашем примере для опции «Тест» можно выбрать сочетание клавиш Alt+Т. Для этого в строке **Caption** указать «&Тест».

Опция «Тест» должна содержать подопции «Тестирование», «Результат», «Выход».

Опция «Помощь» не должна содержать подопций.

Для реализации подопции «Выход» выполните следующие команды:

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/Имя события	Значение/Действие
Выход	Events	OnClick	Close;

3.3. Сохраните форму под именем **F_Menu.pas**.

4. Создадим переход с титульной формы на форму «Меню». Для этого перейдите на титульную форму и создайте процедуры обработки события:

Выделенный объект	Имя события	Действие
«Продолжить» (кнопка)	OnClick	U_Menu.ShowModal ; Комментарий Открывается форма (делается видимой) с помощью метода ShowModal и она становится модальной. Это означает, что управление передается данной форме, и пользователь не может передать фокус другой форме данного приложения до тех пор, пока он не закроет модальную форму.

Кроме того, нужно подсоединить к модулю **F_Titul.pas** форму **U_Menu**, указав в списке подсоединяемых модулей (раздел **Uses**), имя файла **F_Menu**.

```
interface  
  
uses  
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms,  
Dialogs,  
ExtCtrls, StdCtrls, Buttons, F_Menu;
```

Сохраните выполненный проект (**File/Save All**). Запустите программу на выполнение и проверьте правильность ее работы:

- перейдите с титульной формы на форму «Меню»,
- выйти из проекта.

5. Создание формы помощи

5.1. С помощью команды **File/New/Other** откройте окно **New Items**, при помощи которого создайте новую форму (**U_Instr**).

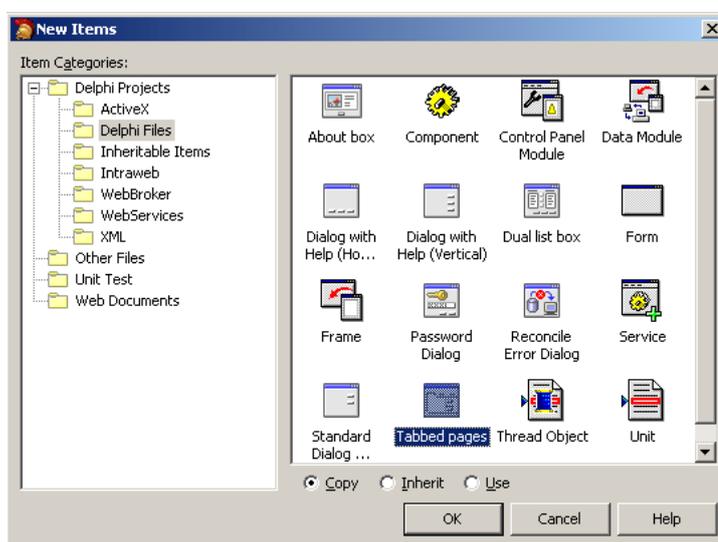


Рис.24

На вкладке **Forms** выберите форму **Tabbed pages**. Появится стандартное окно новой формы **Tabbed Notebook Dialog** (рис.25).

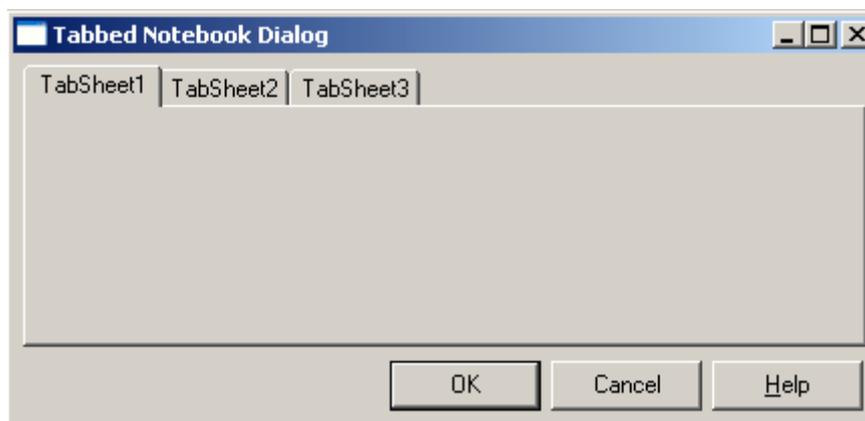


Рис.25

Рассмотрите список объектов, содержащихся на форме (рис.26).

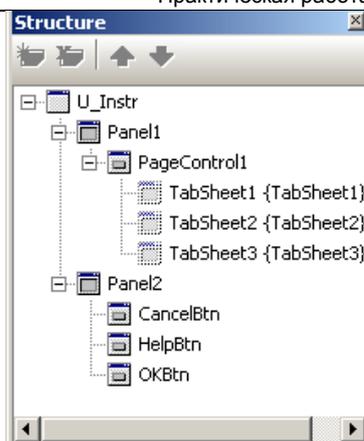


Рис.26

Форма построена на основе компоненты **TPageControl** и содержит несколько перекрывающихся друг друга панелей класса **TTabSheet**. Каждая вкладка может содержать свой набор помещенных на нее компонентов. Чтобы на этапе конструирования добавить новую панель или выбрать ранее вставленную, надо щелкнуть на объекте правой кнопкой мыши и выбрать **New Page** (новая панель), **Next Page** (следующая панель) или **Previous Page** (предыдущая панель). Смена панелей идет циклически.

5.2. Удалите объекты-кнопки «OK» и «Help».

5.3. Замените название кнопки «Cancel» на «Выход» и установите для нее действие `Close`;

5.4. Измените название формы, для чего в **Object Inspector** откройте список объектов, выберите **PagesDlg1**:

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ Имя события	Значение/Действие
PagesDlg1	Properties	Caption	Помощь
		Name	U_Instr установка имени формы, под которым компонент будет известен программе.

5.5. Дайте страницам названия «Правила работы», «Тематика теста», «Авторские права», для этого в свойстве **Caption** замените соответственно **TabSheet1**, **TabSheet2**, **TabSheet3**.

5.6. Заполните страницы формы **Tabbed Notebook Dialog** текстом инструкции, для чего на каждую страницу добавьте экземпляр компонента **Memo** из вкладки **Standard** палитры компонентов. Для ввода текста в объект **Memo** щелкните на кнопке с тремя точками свойства **Lines**. В появившемся окне встроенного редактора ввести необходимый текст.

Для того чтобы вставленный объект совпадал по цвету с формой, выполните следующие действия:

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ Имя события	Значение/Действие
Мемо	Properties	Color	clBtnFace совпадение по цвету с фоном формы
		Ct3D	False отмены трехмерного эффекта объекта
		BorderStyle	bsNone отмены рамки
		Font	выбор нужного шрифта
		Alignment	выравнивания текста

5.7. Сохраните созданную форму под именем **F_Instr.pas**.

5.8. Создайте переход с формы «Меню» на форму «Помощь». Для этого перейдите на форму «Меню», установите курсор на опцию «Помощь», щелкните мышью и внести изменения в процедуру обработки события:

Выделенный объект	Имя события	Действие
Помощь (опция)	OnClick	U_Instr.ShowModal;

Кроме того, нужно подсоединить к модулю **F_Menu.pas** форму **U_Instr**, указав в списке подсоединяемых модулей (раздел Uses), имя файла **F_Instr**.

```
interface

uses
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms,
Dialogs,
Menus, F_Instr;
```

5.9. Сохраните выполненную работу (**File/Save All**). Запустите программу на выполнение и проверите правильность ее работы:

- перейдите с титульной формы на форму «Меню»,
- перейдите с формы «Меню» на форму «Помощь»,
- выйдите из проекта.

6. Создание формы «Тестирование»

6.1. Создайте форму «Тестирование» (**U_Test**) на основе стандартной формы **Tabbed Notebook Dialog** и включите в нее 5 вопросов (5 вкладок **TtabSheet**) как показано на рис. 27.

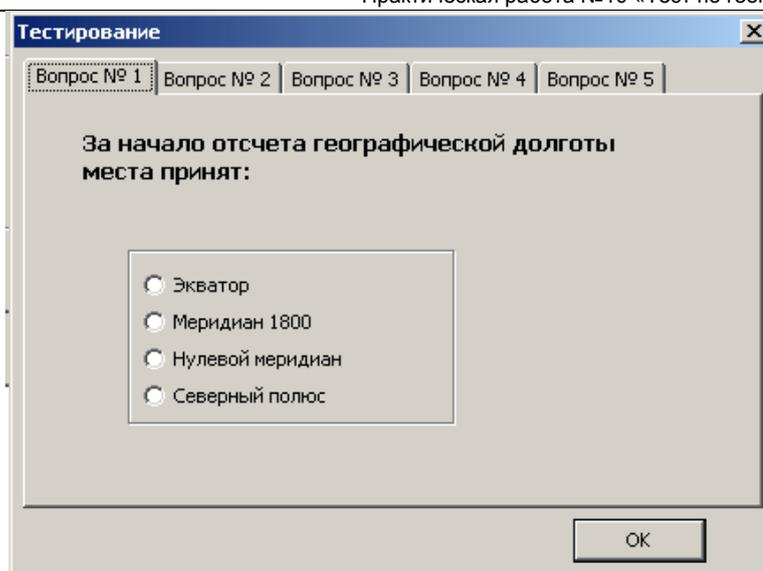


Рис.27

Пример теста по географии:

1) За начало отсчета географической долготы места принят:

- Экватор
- Меридиан 180°
- **Нулевой меридиан**
- Северный полюс

2) Линии, соединяющие на климатических картах точки с одинаковым атмосферным давлением, называются:

- Горизонталями
- Изометрами
- **Изобарами**
- Изогигетрами

3) Артезианскими называются воды:

- Почвенные
- **Грунтовые**
- Межпластовые ненапорные
- Межпластовые напорные

4) Основным географическим следствием шарообразной формы Земли являются:

- Смена дня и ночи
- Смена времени года
- **Широтная зональность**
- Высотная поясность

5) Образование пояса низкого атмосферного давления в экваториальных широтах обусловлено прежде всего:

- Низкими температурами воздуха
- **Высокими температурами воздуха**
- Деятельностью циклонов
- Деятельностью антициклонов

6.2. Самостоятельно создайте форму «Тестирования» и переход на нее из формы «Меню». Для этого повторить все шаги, которые были сделаны при создании формы «Помощь» (см. пункт 5). Сохраните созданную форму под именем **F_Test.pas**.

6.3. Сохраните выполненную работу (**File/Save All**). Запустите программу на выполнение и проверьте правильность ее работы:

- перейдите с «Титульной» формы на форму «Меню»;
- перейдите с формы «Меню» на форму «Тестирования»;
- Выполните тестирования;
- выйти из проекта

7. Создание формы «Результат»

Форма «Результат» (**U_Rez**) должна содержать оценку тестирования и кнопку «Выход».

7.1. Самостоятельно создайте форму «Результат» и переход на нее из формы «Меню». Для этого повторите все шаги, которые были сделаны при создании формы «Помощь» (рис.28).

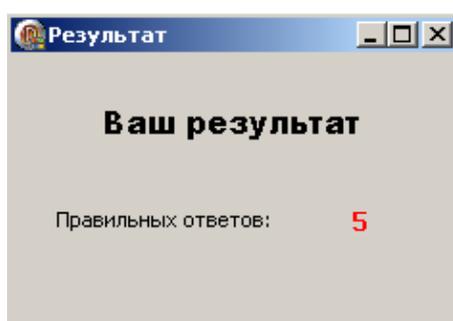


Рис.28

7.2. Для определения правильности ответов объявим типизированную константу **OTV**, которая будет содержать список правильных ответов на все вопросы (правильный ответ – 1, неправильный ответ – 0), а для подсчета количества правильных ответов введем переменную **Ball**. Чтобы **OTV** и **Ball** были доступны в других формах, объявление этих данных необходимо вставить перед разделом **implementation** формы «Результат». Внесите изменения в программу.

```
type  
T1=array[1..5,0..3] of byte;  
const  
OTV:T1=((0,0,1,0),(0,0,1,0),(0,1,0,0),(0,0,1,0),(0,1,0,0));  
var Ball:Byte;  
implementation
```

7.3. Сохраните созданную форму под именем **F_Rez.pas**.

7.4. Для подсчета количества правильных ответов перейдем на форму «Тестирование» и для кнопки «ОК» внести изменения в процедуру обработки события:

Выделенный объект	Имя события	Действие
ОК	OnClick	<pre>Ball:=otv[1,RadioGroup1.ItemIndex]+ otv[2,RadioGroup2.ItemIndex]+ otv[3,RadioGroup3.ItemIndex]+ otv[4,RadioGroup4.ItemIndex]+ otv[5,RadioGroup5.ItemIndex];</pre> Комментарий Суммируются значения констант,

		<p>определяемые индексом переключателей Radiogroup для каждого выбранного ответа на вопрос.</p> <p>Кроме того, нужно описать в модуле F_Test.pas внешнее имя F_Rez, разместив его в списке подключаемых модулей:</p> <pre>interface uses Windows, SysUtils, Classes, Graphics, Forms, Controls, StdCtrls, Buttons, ComCtrls, ExtCtrls, F_Rez;</pre>
--	--	--

7.5. Создайте переход с формы «Меню» на форму «Результат». Для этого нужно перейдите на форму «Меню» и установите курсор на опции «Результат», щелкните мышью и внесите изменения в процедуру обработки:

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/Имя события	Значение/Действие
Результат (опция)	Events	OnClick	<pre>U_Rez.Label3.Caption:= IntToStr(Ball); U_Rez.ShowModal;</pre> <p>Кроме того, нужно описать в модуле F_Menu.pas внешнее имя F_Rez, разместив его в списке подключаемых модулей:</p> <pre>interface uses Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs, Menus, F_Instr, F_Test, F_Rez;</pre>

8. Проверьте правильность работы программы.

Задание для самостоятельного выполнения

	Задание
1	Дополните программу разделом «Содержание изучаемой темы».
2	Внесите изменения в программу так, чтобы в начале запрашивалось Имя тестируемого, которое выводилось бы на экран при выдаче результата прохождения теста.
3	Для наглядности выводите результат разным цветом.
4	Усовершенствуйте программу, введя в раздел описания массив констант с номерами правильных ответов, а в программе только сравнивайте индекс переключателя со значением константы.
5	Введите дополнительную кнопку Повторить , которая позволяет повторно Выполните задание, указав новое имя.

6	Создайте тестирующую программу по истории, литературе, математике или какому-либо другому предмету. Тест должен содержать не менее 4 разделов, каждый из которых должен включать не менее 15 вопросов.
---	--

Листинг программы

Для самоконтроля ниже приводится сводная таблица форм и текст программы.

Название форм	Имя файла *.PAS	Свойство Name	Подсоединяемые формы (Uses)
Титульная форма	F_Titul.pas	U_Titul	F_Menu
Меню	F_menu.pas	U_menu	F_Instr, F_test, F_Rez
Тестирование	F_test.pas	U_test	F_Rez
Помощь	F_Instr.pas	U_Instr	
Результат	F_Rez.pas	U_Rez	

Титульная форма

```

unit F_Titul;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Buttons, jpeg, ExtCtrls, F_Menu;
type
  TU_Titul = class(TForm)
    Image1: TImage;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    Label1: TLabel;
    Label2: TLabel;
    procedure BitBtn2Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  U_Titul: TU_Titul;
implementation
{$R *.dfm}
procedure TU_Titul.BitBtn2Click(Sender: TObject);
begin
  U_Menu.ShowModal;
end;
end.
    
```

Форма «Меню»

```
unit F_menu;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Menus, F_Instr, F_test, F_Rez;
type
  TU_Menu = class(TForm)
    MainMenu1: TMainMenu;
    N1: TMenuItem;
    N2: TMenuItem;
    N3: TMenuItem;
    N4: TMenuItem;
    N5: TMenuItem;
    procedure N5Click(Sender: TObject);
    procedure N2Click(Sender: TObject);
    procedure N3Click(Sender: TObject);
    procedure N4Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  U_Menu: TU_Menu;
implementation
{$R *.dfm}
procedure TU_Menu.N2Click(Sender: TObject);
begin
  U_Instr.ShowModal;
end;

procedure TU_Menu.N3Click(Sender: TObject);
begin
  U_Test.RadioGroup1.ItemIndex:=-1;
  U_Test.RadioGroup2.ItemIndex:=-1;
  U_Test.RadioGroup3.ItemIndex:=-1;
  U_Test.RadioGroup4.ItemIndex:=-1;
  U_Test.RadioGroup5.ItemIndex:=-1;
  U_Test.PageControl1.ActivePage:=U_Test.TabSheet1;
  U_Test.ShowModal;
end;
procedure TU_Menu.N4Click(Sender: TObject);
begin
  U_Rez.Label3.Caption:=IntToStr(Ball);
  U_Rez.ShowModal;
end;
procedure TU_Menu.N5Click(Sender: TObject);
begin
  Close;
end;
end.
```

Форма «Помощь»

```
unit F_Instr;

interface

uses Windows, SysUtils, Classes, Graphics, Forms, Controls, StdCtrls,
    Buttons, ComCtrls, ExtCtrls;

type
    TU_Instr = class(TForm)
        Panel1: TPanel;
        Panel2: TPanel;
        PageControl1: TPageControl;
        TabSheet1: TTabSheet;
        TabSheet2: TTabSheet;
        TabSheet3: TTabSheet;
        CancelBtn: TButton;
        procedure CancelBtnClick(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    U_Instr: TU_Instr;

implementation

{$R *.dfm}

procedure TU_Instr.CancelBtnClick(Sender: TObject);
begin
    Close;
end;

end.
```

Форма «Тестирование»

```
unit F_test;

interface

uses Windows, SysUtils, Classes, Graphics, Forms, Controls, StdCtrls,
    Buttons, ComCtrls, ExtCtrls, F_Rez;

type
    TU_Test = class(TForm)
        Panel1: TPanel;
        Panel2: TPanel;
        PageControl1: TPageControl;
        TabSheet1: TTabSheet;
        TabSheet2: TTabSheet;
        TabSheet3: TTabSheet;
        OKBtn: TButton;
        TabSheet4: TTabSheet;
        TabSheet5: TTabSheet;
        Label1: TLabel;
        RadioGroup1: TRadioGroup;
        Label2: TLabel;
        RadioGroup2: TRadioGroup;
        Label3: TLabel;
        RadioGroup3: TRadioGroup;
        Label4: TLabel;
        RadioGroup4: TRadioGroup;
        Label5: TLabel;
        RadioGroup5: TRadioGroup;
        procedure OKBtnClick(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    U_Test: TU_Test;

implementation

{$R *.dfm}

procedure TU_Test.OKBtnClick(Sender: TObject);
begin
    Ball:=otv[1, RadioGroup1.ItemIndex]+
    otv[2, RadioGroup2.ItemIndex]+
    otv[3, RadioGroup3.ItemIndex]+
    otv[4, RadioGroup4.ItemIndex]+
    otv[5, RadioGroup5.ItemIndex];

end;

end.
```

Форма «Результат»

```
unit F_Rez;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls;

type
  TU_Rez = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  U_Rez: TU_Rez;
type
  T1=array[1..5,0..3] of byte;
const
  otv:T1=((0,0,1,0),(0,0,1,0),(0,1,0,0),(0,0,1,0),(0,1,0,0));
Var Ball:Byte;
implementation

{$R *.dfm}

procedure TU_Rez.FormCreate(Sender: TObject);
begin

Label3.Caption:=IntToStr(Ball);
end;

end.
```