

Створення простого теста

На цьому уроці:

Ми навчимося створювати і перевіряти умови, користуватися компонентами Radio Button і Checkbox і складемо просту тестуючу програму

Питання:

1. Розглянемо використання компонентів Delphi **Radio Button** і **Checkbox**
2. Створення простої тестуючої програми

А зараз давайте спробуємо, використовуючи компоненти **Radio Button і **Check Box** створити простий тест з мінімальними вимогами до програми (обійдемося поки додатком з однієї форми, без реєстрації, окремих файлів тестів, картинок і ін.) – зробимо це по кроках**

КРОК 1

Звичайно, спочатку потрібно скласти тест на папері. Для прикладу візьмемо тест з 5 питань, причому перші три питання з єдиним вибором, а останні два – з множинним:

з єдиним вибором

1	Верно ли, что вся информация в компьютере хранится в двоичном коде	Только числовая
		Только числовая и текстовая
		Любая информация
		Вся информация за исключением видео
2	Наименьшая единица информации называется	байт
		бит
		бод
		бит/с
3	Элементарной базой первых компьютеров являлись	микросхемы
		дискретные полупроводниковые элементы
		радиолампы
		транзисторы

з множинним вибором

4	Назовите устройства вывода информации	Монитор
		мышь
		клавиатура
		принтер
5	Назовите системы программирования	Delphi
		Visual Basic
		Microsoft
		Turbo Pascal

КРОК 2

Зараз давайте визначимо вимоги до програми :

Для першої тестуючої програми ми обмежимося тільки наступними функціями:

Підрахунок числа вірних відповідей

Підрахунок % вірних відповідей

Виведення результату після закінчення тесту

Блокування повернення до попередніх питань

Використання 4 варіантів вибору відповіді

Використання завдань як з єдиним, так і множинним вибором відповіді

Після кожної відповіді програма повинна оповіщати нас про правильність нашого вибору

І на цьому функціональність програми поки обмежимо

КРОК 3

Наступним кроком буде розробка зовнішнього вигляду додатка і визначення компонент, яких ми використовуємо для введення і виведення інформації

The screenshot shows a Windows Forms application window titled "Form1" with a standard XP-style title bar. The form is laid out on a dotted grid background and contains five distinct sections, each enclosed in a "GroupBox":

- GroupBox1 (top-left):** Contains "Label1", a "RadioGroup1" with four radio buttons, "Label6", and "Button1".
- GroupBox2 (middle-left):** Contains "Label2", a "RadioGroup2" with four radio buttons, "Label7", and "Button2".
- GroupBox3 (bottom-left):** Contains "Label3", a "RadioGroup3" with four radio buttons, "Label8", and "Button3".
- GroupBox4 (top-right):** Contains "Label4", four "CheckBox" controls (labeled "CheckBox1" through "CheckBox4"), "Label9", and "Button4".
- GroupBox5 (middle-right):** Contains "Label5", four "CheckBox" controls (labeled "CheckBox5" through "CheckBox8"), "Label10", and "Button5".

At the bottom of the form, there is a text box labeled "Edit1" and a "Button6".

Розміщуємо для кожного питання свій компонент Group Box

Розміщуємо Label для виведення тексту питання

Розміщуємо Radio Group на 4 варіанта для відповіді!

КРОК 4

Наступним кроком буде розробка зовнішнього вигляду додатка і визначення компонент, яких ми використовуватимемо для введення і виведення інформації

The screenshot shows a Windows Forms application window titled "Form1" with a standard Windows XP-style title bar. The form contains five group boxes arranged in a grid:

- GroupBox1**: Contains Label1, a RadioGroup1 with five radio buttons, Label6, and Button1. Label6 and Button1 are highlighted with red rectangles.
- GroupBox2**: Contains Label2, a RadioGroup2 with five radio buttons, Label7, and Button2.
- GroupBox3**: Contains Label3, a RadioGroup3 with five radio buttons, Label8, and Button3.
- GroupBox4**: Contains Label4, four checkboxes (CheckBox1-4), Label9, and Button4.
- GroupBox5**: Contains Label5, four checkboxes (CheckBox5-8), Label10, and Button5.

At the bottom of the form, there is a large empty rectangular area and Button6.

Розміщуємо компонент Label для визначення зовнішнього вигляду (він/и) навірно ми (виповіли)

Розміщуємо кнопку для перевірки вибраної відповіді!

КРОК 5

Наступним кроком буде розробка зовнішнього вигляду додатка і визначення компонент, яких ми використовуватимемо для введення і виведення інформації

The screenshot shows a Windows Forms application window titled "Form1" with a standard XP-style title bar. The form contains five grouped boxes (GroupBox1 to GroupBox5) arranged in a grid. Each box contains a label, a radio group, and a button. The second box from the top left (GroupBox2) is highlighted with a red border. The layout is as follows:

- GroupBox1:** Label1, RadioGroup1, Label6, Button1.
- GroupBox2:** Label2, RadioGroup2, Label7, Button2. (Highlighted with a red border)
- GroupBox3:** Label3, RadioGroup3, Label8, Button3.
- GroupBox4:** Label4, CheckBox1, CheckBox2, CheckBox3, CheckBox4, Label9, Button4.
- GroupBox5:** Label5, CheckBox5, CheckBox6, CheckBox7, CheckBox8, Label10, Button5.

At the bottom center of the form, there is a large empty text box and a Button6.

Аналогічно
формлюємо
2 та 3
питання

КРОК 6

Наступним кроком буде розробка зовнішнього вигляду додатка і визначення компонент, яких ми використовуватимемо для введення і виведення інформації

The screenshot shows a Windows Forms application window titled "Form1" with a standard Windows XP-style title bar. The form contains several controls arranged in a grid-like fashion:

- GroupBox1:** Contains Label1, RadioGroup1 (with five radio buttons), Label6, and Button1.
- GroupBox2:** Contains Label2, RadioGroup2 (with five radio buttons), Label7, and Button2.
- GroupBox3:** Contains Label3, RadioGroup3 (with five radio buttons), Label8, and Button3.
- GroupBox4:** Contains Label4, four CheckBoxes (labeled CheckBox1 through CheckBox4), Label9, and Button4. This group box is highlighted with a red border.
- GroupBox5:** Contains Label5, four CheckBoxes (labeled CheckBox5 through CheckBox8), Label10, and Button5. This group box is also highlighted with a red border.
- GroupBox6:** A large empty rectangular area, highlighted with a red border.
- Button6:** A single button located at the bottom right of the form, also highlighted with a red border.

Для 4 і 5 питань
(можлиний видір)
замість
радіокнопок
вставляємо по 4
Check Box

Розміщуємо
компонент Memo
для виведення
результатів теста

Розміщуємо,
кнопку для
завершення роботи
з програмою

КРОК 7

Тест по информатике

Вопрос 1
Верно ли, что информация в компьютере хранится в двоичном коде

Варианты ответов

Только числовая

Только числовая и текстовая

Любая информация

Вся за исключением видео

Label6

проверить

Вопрос 2
Наименьшая единица информации называется

Варианты ответов

байт

бит

бод

бит/с

Label7

проверить

Вопрос 3
Элементарной базой первых компьютеров являлись

Варианты ответов

Микросхемы

Дискретные полупроводники

Радиолампы

Транзисторы

Label8

проверить

Вопрос 4
Назовите устройства вывода информации

Монитор

Мышь

Клавиатура

Принтер

Label9

проверить

Вопрос 5
Назовите системы программирования

Delphi

Visual Basic

Microsoft

Turbo Pascal

Label10

проверить

Memo1

ВЫХОД

В результаті ми отримали прикладну форму, на якій є питання, варіанти відповіді, Memo для введення результату, кнопки перевірки і виходу

КРОК 8

Додаток оформлений. Почнемо програмування подій. І найперша подія, яка виникає кожного разу при запуску програми, – створення форми (**On Create**)

Що повинне відбуватися при запуску програми?

Тест по информатике

Вопрос 1
Верно ли, что информация в компьютере хранится в двоичном коде

Варианты ответов

- Только числовая
- Только числовая и текстовая
- Любая информация
- Вся за исключением видео

проверить

Label6

Вопрос 2
Наименьшая единица информации называется

Варианты ответов

- байт
- бит
- бод
- бит/с

проверить

Label7

Вопрос 3
Элементарной базой первых компьютеров являлись

Варианты ответов

- Микросхемы
- Дискретные полупроводники
- Радиолампы
- Транзисторы

проверить

Label8

Вопрос 4
Назовите устройства вывода информации

- Монитор
- Мышь
- Клавиатура
- Принтер

проверить

Label9

Вопрос 5
Назовите системы программирования

- Delphi
- Visual Basic
- Microsoft
- Turbo Pascal

проверить

Label10

Memo1

ВЫХОД

1. Мітки (**Label 6,7,8,9,10**) повинні бути невидимі
2. **Мемо** теж повинно бути невидим і з'являється тільки після проходження теста
3. Лічильник **k**, який рахує вірно виконані завдання, повинен обнулятися

КРОК 9

Додаток оформлений. Почнемо програмування подій. І найперша подія, яка виникає кожного разу при запуску програми, – створення форми (**On Create**)

Зробимо подвійне клацання формою і в редакторі коди запишемо процедуру створення форми (**On Create**)

```
var
  Form1: TForm1;
  k: integer;
implementation
  {$R *.dfm}

  procedure TForm1.FormCreate(Sender: TObject);
  begin
    Memo1.Visible:=false;
    Memo1.Text:='';
    Label6.Visible:=false;
    Label7.Visible:=false;
    Label8.Visible:=false;
    Label9.Visible:=false;
    Label10.Visible:=false;
    k:=0;
  end;
```

Оголосуємо змінну **k**, яка є
личильником правильних
відповідей

Робимо невидимими
Label – и ! Memo

Личильнику **k**
присвоюємо значення нуль

КРОК 10

Зараз напишемо програму на подію натиснення на кнопку «**ПЕРЕВІРИТИ**»

Зробимо подвійне клацання по кнопці «**ПЕРЕВІРИТИ**» в першому питанні і запишемо відповідний код.

```
Unit1
procedure TForm1.Button1Click(Sender: TObject);
begin
  Button1.Enabled:=false;
  if radiogroup1.ItemIndex=2 then
  begin
    k:=k+1;
    label6.Font.Color:=rgb(0,150,0);
    label6.Visible:=true;
    label6.Caption:='ВЕРНО !!';
  end
  else
  begin
    label6.Font.Color:=rgb(150,0,0);
    label6.Visible:=true;
    label6.Caption:='НЕВЕРНО !!';
  end;
end;
```

Давайте його проаналізуємо

Кнопку «ПЕРЕВІРИТИ» в першому питанні робимо недоступною, після того, як вона натискається. Це унеможливіло повернення до першого питання і виводу іншої відповіді!

Перевіримо умову: чи відповідає наш вибір правильному, тобто чи вибраний перемикач з індексом 2



Відмітьте ! Індеси перемикачів в Delphi мають нумерацію, що починається з нуля, тому вірна відповідь у нас позначена індексом 2, хоча насправді це третій варіант відповідей

КРОК 11

Зараз напишемо програму на подію натиснення на кнопку «**ПЕРЕВІРИТИ**»

Зробимо подвійне клацання по кнопці «**ПЕРЕВІРИТИ**» в першому питанні і запишемо відповідний код.

```
Unit1
procedure TForm1.Button1Click(Sender: TObject);
begin
  Button1.Enabled:=false;
  if radiogroup1.ItemIndex=2 then
  begin
    k:=k+1;
    label16.Font.Color:=rgb(0,150,0);
    label16.Visible:=true;
    label16.Caption:='ВЕРНО !!';
  end
  else
  begin
    label16.Font.Color:=rgb(150,0,0);
    label16.Visible:=true;
    label16.Caption:='НЕВЕРНО !!';
  end;
end;
```

Давайте його проаналізуємо

Якщо умова виконується (вибрана вірна відповідь), то значення лічильника вірних відповідей збільшимо на одиницю.
Мітку b(Label6), де виводиться Вірно/Невірно робимо видимою і пишемо в ній напис **ВІРНО** зеленим кольором

Якщо умова не виконується, то лічильник не збільшимо, а мітку b робимо видимою і виводимо напис **НЕВІРНО**, причому червоним кольором



Очевидно, що для кнопок перевірки 2 і 3 питань код буде абсолютний таким же за винятком номерів Label-ів для виведення ВІРНО/НЕВІРНО, тому Ви можете написати його самостійно (не забувайте, що в редакторві можна використовувати копіювання коду – це набагато швидше)

КРОК 12

Наступним кроком опишемо процедури перевірки відповіді в питаннях 4 і 5 – там код буде трохи іншої зважаючи на множинний вибір і інших використовуваних компонент

Зробимо подвійне клацання по кнопці «**ПЕРЕВІРИТИ**» в 4 питанні і запишемо відповідний код.

```

procedure TForm1.Button4Click(Sender: TObject);
begin
    Button4.Enabled:=false;
    if (checkbox1.Checked=true) and (checkbox4.Checked=true)
    and (checkbox2.Checked=false) and (checkbox3.Checked=false) then
    begin
        k:=k+1;
        label19.Font.Color:=rgb(0,150,0);
        label19.Visible:=true;
        label19.Caption:='ВЕРНО !!'
    end
    else
    begin
        label19.Font.Color:=rgb(150,0,0);
        label19.Visible:=true;
        label19.Caption:='НЕВЕРНО !!'
    end;
end;

```

Перевіряємо вибрані відповіді: відповідь буде вірною тільки тоді, коли перший і четвертий бокси будуть вибрані (True), а другий і третій не вибрані (false)

У цьому випадку значення лічильника вірних відповідей збільшимо на одиницю
Мітку (Label) робимо видимою і пишемо в ній напис **ВЕРНО** зеленим кольором

Якщо умова не виконується, то лічильник не збільшимо, мітку 9 робимо видимою і виводимо напис **НЕВЕРНО** червоним кольором

КРОК 13

Хоча **питання 5** теж з множинним вибором, то код обробки цієї кнопки аналогічний кнопці 4, але давайте додамо в неї додаткові функції: після натиснення цієї кнопки повинні виводитися результати тесту в **Мемо**

```
procedure TForm1.Button5Click(Sender: TObject);
begin
```

```
    Button5.Enabled:=false;
    if (checkbox5.Checked=true) and (checkbox6.Checked=true)
    and (checkbox8.Checked=true) and (checkbox7.Checked=false) then
begin
    k:=k+1;
    label10.Font.Color:=rgb(0,150,0);
    label10.Visible:=true;
    label10.Caption:='ВЕРНО !!'
end
else
begin
    label10.Font.Color:=rgb(150,0,0);
    label10.Visible:=true;
    label10.Caption:='НЕВЕРНО !!';
end;
```

```
memo1.Visible:=true;
```

```
memo1.Text:='Всего вопросов - 5'+#13+#10+'Количество верных ответов - '
+inttostr(k)+#13+#10+'Процент выполнения теста - '+floattostr(k/5*100)+' %';
```

Ця частка аналогічна
кнопці 4 питання і поясненн
я не вимагається

Робимо компонент **Мемо**
видимим для введення
результатів

Виводимо результати тесту

КРОК 14Давайте подивимось, як виводиться в **Мемо** результат теста

```
memo1.Text:='Всього вопросов - 5'+#13+#10+'Количество верных ответов - '
+inttostr(k)+#13+#10+'Процент выполнения теста - '+floattostr(k/5*100)+' %';
```

Всього питань - 5

В апострофах (одинарних
лапках) виводиться рядок
символів:

Кількість правильних відповідей

Виведення тексту

Це означає, що виведення
елементу початку рядка
початку наступного рядка

Давайте пригадаємо таблицю
кодування символів - ASCII -
символи з номерами 13 і 10
зарезервовані для перенесення
курсора на наступний рядок (Enter) і в
!! початок відповідно

БАЗОВАЯ ТАБЛИЦА ASCII

3	4	5	6	7	8	9	A	B	C	D	E	F
3	4	5	6	7	8	9	10	11	12	13	14	15
!			-		↑	↓	→	←	↔	^	▼	
19	20	21	22	23	24	25	26	27	28	29	30	31
#	\$	%	&	'	()	*	+	,	-	.	/
35	36	37	38	39	40	41	42	43	44	45	46	47
3	4	5	6	7	8	9	:	;	<	=	>	?
51	52	53	54	55	56	57	58	59	60	61	62	63

КРОК 15

Давайте подивимось, як виводиться в **Мемо** результат теста

```
memo1.Text:='Всього вопросов - 5'+#13+#10+'Количество верных ответов - '
+inttostr(k)+#13+#10+'Процент выполнения теста - '+floattostr(k/5*100)+' %';
```

Переводимо курсор на новий рядок

Рахуємо процент виконання! переводимо його в рядкову величину (із дійсного числа)

Виводимо символ %

Виводимо значення лічильника k , переводимо його значення в рядкову величину (`inttostr`)

Виводимо текст
Процент виконання
теста

А ось результат, виведений в Мемо

```
Всього вопросов - 5
Количество верных ответов - 3
Процент выполнения теста - 60 %
```

КРОК 16

Остання кнопка – ВИХІД

```
procedure TForm1.Button6Click(Sender: TObject);  
begin  
close  
end;
```

Закриваємо

КРОК 17

Зберігаємо і компілюємо програму
Запускаємо наш тест

Запустити ->

На цьому наш урок закінчений.

Звичайно, наша тестуюча програма вийшла досить примітивна, у неї відсутні багато необхідних функцій. У міру вивчення Delphi ми складемо серйознішу і зручнішу тестуючу оболонку, а поки найголовніше, що наша програма працює і ми навчилися використовувати компоненти для перевірки умов

Висновки:

На цьому уроці ми навчилися створювати і перевіряти умови, користуватися компонентами **Radio Button** і **Checkbox** і склали просту тестуючу програму
