

## Практическая работа №6, ЦВЕТА В ФОРМАТЕ RGB

### Постановка задачи



Создайте программу, с помощью которой пользователь может увидеть в зависимости от значений насыщенности красного, зеленого и синего результирующий цвет (рис.6.1.).

Рис.6.1.

### Новым в этой работе являются:

- использование для ввода данных полосы прокрутки **ScrollBar** вкладки палитры компонентов **Standard**,
- компонента панель **Panel** вкладки палитры компонентов **Standard**,
- функция преобразования значений цветовых составляющих – **TColorRef**.

### План разработки программы

1. Откройте новый проект.
2. Разместите в форме экземпляры компонентов в соответствии с рис.6.2.

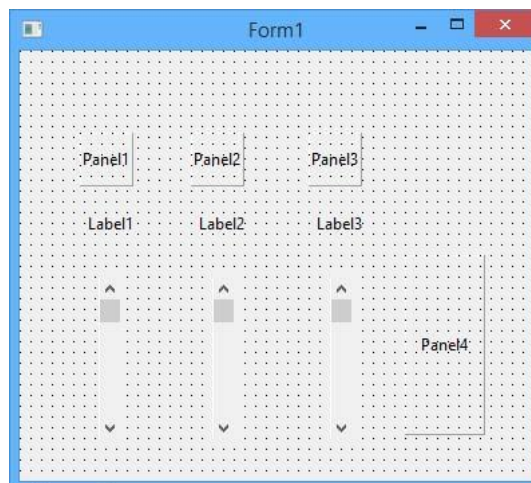


Рис.6.2.

### Комментарий

Полоса прокрутки **ScrollBar** может быть горизонтальной (по умолчанию) или вертикальной. Это определяется свойством **Kind**. В нашем случае используется вертикальная полоса прокрутки.

3. Сохраните код программы и проект под именами, например, **fMain.pas** и **Color.lpr**.

4. Выполните следующие действия:

Выделенный объект	Вкладка окна Инспектор объектов	Имя свойства/Имя события	Значение/Действие
Panel1	Properties	Name	RedPanel <b>Комментарий</b> Установка имени панели <b>RedPanel</b> , под которым компонент будет известен программе.
		Caption	Удалить название компонента
Label1	Properties	Name	RedLabel <b>Комментарий</b> Установка имени метки <b>RedLabel</b> , под которым компонент будет известен программе.
		Caption	Удалить название компонента
ScrollBar1	Properties	Name	RedBar <b>Комментарий</b> Установка имени полосы прокрутки <b>RedBar</b> , под которым компонент будет известен программе.
		Max	255 <b>Комментарий</b> Максимальный диапазон целых значений – количество градаций компонента RGB.
		Position	122 <b>Комментарий</b> Начальная позиция ползунка – начальное значение.

Аналогично задайте значения для **ScrollBar2, Panel2, Label2**, присвоив им имена **GreenBar, GreenPanel, GreenLabel** и **ScrollBar3, Panel3, Label3**, присвоив им имена **BlueBar, BluePanel, BlueLabel**.

5. Когда на форме много компонентов, ручное выравнивание становится весьма утомительным занятием. Для этого случая в среде Delphi предусмотрены средства автоматизированного выравнивания компонентов.

Повторите эту же операцию для других групп компонент (**GreenBar, GreenPanel, GreenLabel** и **BlueBar, BluePanel, BlueLabel**).

6. Выполните следующие действия:

Выделенный объект	Вкладка окна Инспектор объектов	Имя свойства/ Имя события	Значение/Действие
RedBar (см. п.4 - ScrollBar1)	Events	OnChange	<pre>RedPanel.Color:= ColorToRGB (RGBToColor (RedBar.Position,0,0)); RedLabel.Caption:=IntToStr (RedBar.Position); Panel4.Color:= ColorToRGB (RGBToColor (RedBar.Position, GreenBar.Position, BlueBar.Position));</pre> <p><b>Комментарий</b></p> <p>В зависимости от передвижения ползунка <b>RedBar</b>, будет меняться цвет панели <b>RedPanel</b>, выводиться числовое значение кода на месте <b>RedLabel</b> и меняться цвет панели <b>Panel4</b>.</p> <p>Функция RGBToColor (R,G,B) превращает три составляющие цвета из трех отдельных значений в одно целое значение цвета. У этой функции три параметра R – значение красного цвета, G – значение зеленого цвета, B – значение синего цвета.</p> <p>В нашем случае в качестве параметров используются значения соответствующих полос прокрутки <b>ScrollBar</b>.</p> <p><b>ColorToRGB</b> – это 32-битовое значение, соответствующее цвету, которое получается с помощью функции RGB.</p>

Аналогично задайте значения для **GreenBar** и **BlueBar**.

Проследите за правильностью записи параметров в функции **RGBToColor**.

7. Сохраните проект окончательно, запустите и протестируйте его.

**Задание для самостоятельного выполнения**

1. Усовершенствуйте проект:

- a) сделайте к программе заголовок,
- b) вставьте кнопку выхода из программы.

2. Предусмотрите, чтобы при запуске программы были установлены начальные цвета панелей в зависимости от исходных значений ползунков.

*Подсказка.* Поместите в событие при создании формы (**OnCreate** для Form1) обработку значений позиции **RedBar**, **GreenBar** и **BlueBar**.