

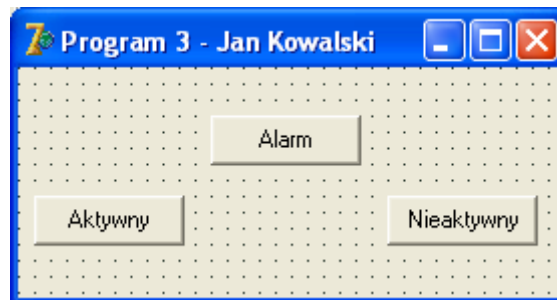
## Ćwiczenie 3 - Edycja właściwości komponentów

Ćwiczenie ma za zadanie naukę utrwalenie umiejętności edycji właściwości komponentów. Poznanie właściwości *Enabled*. Wstawianie dymku podpowiedzi (*Hint*) i edycja kolorów.

### 1. Utworzenie nowego projektu

*Enabled* to właściwość, która mówi czy element ma być aktywny, czy nie.

1. Należy utworzyć nowy projekt (PROJEKT->NOWY PROJEKT->APLIKACJA), jeśli jeszcze nie jest utworzony.
2. Na formatkę proszę wstawić trzy przyciski **BUTTON** z zakładki **STANDARD**.
3. Przycisk **BUTTON1** ma mieć napis (zmień właściwość **CAPTION**) **Alarm**. Po jego naciśnięciu ma się pojawić komunikat **Alarm**.
4. Wszystkie przyciski rozmieść równomiernie. Dostosuj do nich wielkość formatki **FORM1**. Jako tytuł programu wpisz **Program 3** i swoje imię i nazwisko.
5. Całość ma wyglądać następująco:



### 2. Edycja właściwości komponentu **BUTTON1** - *Enabled*

1. Należy zmienić napis na przycisku **BUTTON2** na **Aktywny** (właściwość **CAPTION**).
2. Następnie wybrać jego zdarzenie **OnClick**.
3. W kodzie programu pojawi się nowa procedura. Wprowadź do niej linijkę:

```
procedure TForm1.Button2Click(Sender: TObject);  
begin  
  Button1.Enabled:=true;  
end;
```

4. Podobnie trzeba zrobić z przyciskiem **BUTTON3**. Zmień napis na nim na **Nieaktywny** (właściwość **CAPTION**).
5. Następnie wybrać jego zdarzenie **OnClick**.
6. W kodzie programu pojawi się nowa procedura. Wprowadź do niej linijkę:

```
procedure TForm1.Button3Click(Sender: TObject);  
begin  
  Button1.Enabled:=false;  
end;
```

7. Dodaj przycisk **BUTTON** zamykający program. Napis na nim to **Zamknij**.
8. Program ma mieć wygląd:



9. Skompiluj program. Usuń ewentualne błędy. Uruchom i przetestuj działanie aplikacji.

### 3. Edycja właściwości komponentu BUTTON1 - Hint

*Hint* to podpowiedź (dymek), która ukazuje się gdy najedziemy myszą na dany komponent. Składają się na nie dwie właściwości komponentu:

- **Hint** – zawiera tekst, który ma się wyświetlać
  - **Showhint**
    - **true** pozwala na wyświetlenie podpowiedzi,
    - **false** nie wyświetla.
1. Dodaj następujące podpowiedzi do wcześniejszego programu.
    - a. Do przycisku `Alarm` – „Wyświetla komunikat alarmowy”
    - b. Do przycisku `Aktywny` – „Uaktywnia przycisk Alarm”
    - c. Do przycisku `Nieaktywny` – „Blokuję przycisk Alarm”
    - d. Do przycisku `Zamknij` – „Zamyka program”
  2. Dla każdego przycisku wpisz ten tekst we właściwość **hint**. Właściwość **showhint** ustaw na **true**.
  3. Skompiluj program. Usuń ewentualne błędy. Uruchom i przetestuj działanie aplikacji.

### 4. Kolory w Delphi

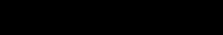



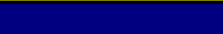















Zmiana kolorów może się odbywać poprzez zmianę właściwości komponentu `COLOR`. Można też dokonać tego poprzez odpowiednie polecenia w kodzie programu. Zmienna koloru ma postać typu **clred**, gdzie **cl** to znacznik zmiennej koloru.

```
procedure TForm1.Button3Click(Sender: TObject);
begin
  Label1.Color:=clRed;
end;
```

Zamiast nazw kolorów można też użyć wartości RGB dla danego koloru.

```
procedure TForm1.Button3Click(Sender: TObject);
begin
  Label1.Color:= $0000FF;
end;
```

Tabela 1 - Lista standardowych kolorów w Delphi

Numer koloru	Wartość RGB koloru	Kolor	Nazwa	Definicja koloru w Delphi
0	\$000000		Czarny	clBlack
1	\$000080		Ciemnoczerwony	clMaroon
2	\$008000		Zielony	clGreen
3	\$008080		Oliwkowy	clOlive
4	\$800000		Ciemnoniebieski	clNavy
5	\$800080		Fioletowy	clPurple
6	\$808000		Ciemnozielony	clTeal
7	\$C0C0C0		Srebrny	clSilver
8	\$C0DCC0		Bladozielony	clMoneyGreen
9	\$F0CAA6		Błękitny	clSkyBlue
246	\$F0FBFF		Kremowy	clCream
247	\$A4A0A0		Jasnoszary	clMedGray
248	\$808080		Szary	clGray, cl DkGray
249	\$0000FF		Czerwony	clRed
250	\$00FF00		Jasnozielony	clLime
251	\$00FFFF		Żółty	clYellow
252	\$FF0000		Niebieski	clBlue
253	\$FF00FF		Różowy	clFuchsia
254	\$FFFF00		Jasnoniebieski	clAqua
255	\$FFFFFF		Biały	clWhite

## 5. Edycja kolorów komponentu Label1

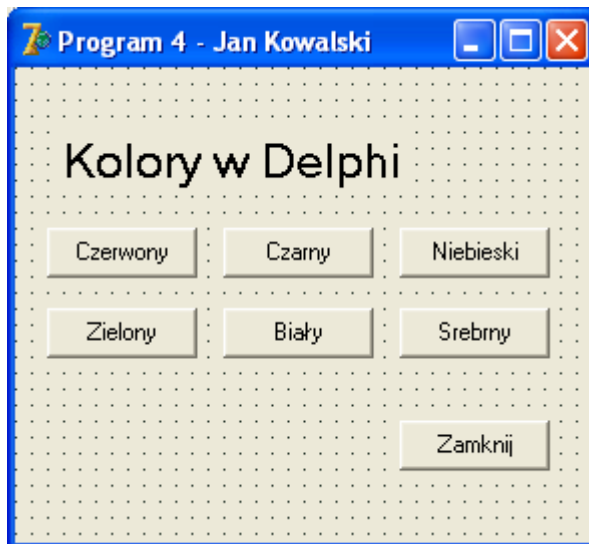
1. Utwórz nowy projekt. Wstaw etykietę LABEL1.
  - a. Napis na etykiecie (właściwość CAPTION) zmień na Kolory w Delphi.
  - b. Własność AUTOSIZE ustaw na **false**.
  - c. Właściwości czcionki (FONT) ustaw następująco:
    - i. Wielkość czcionki (HEIGHT) = 30
2. Wstaw 6 przycisków BUTTON.
  - a. Zmień napisy na nich, na nazwy kolorów: Czerwony, Zielony, Czarny, Biały, Niebieski, Srebrny.
  - b. Rozmieść je w 3 kolumnach (jak na rysunku).
  - c. Do zdarzenia ONCLICK każdego z nich wstaw następujące polecenie:

```

procedure TForm1.Button1Click(Sender: TObject);
begin
    Label1.Color:= clWhite;
end;

```

3. Dodaj przycisk zamykający program.
4. Program ma wyglądać następująco:



5. Skompiluj program. Usuń ewentualne błędy. Uruchom i przetestuj działanie aplikacji.

## ***Zadania***

### **Zadanie 3.1**

Napisz program, który zmienia kolor formatki FORM1, na inny. Nazwa danego koloru ma być napisana na przycisku BUTTON.

Utwórz przyciski z następującymi kolorami:

- Czerwony
- Zielony
- Niebieski
- Czarny
- Biały
- Żółty

Dodaj podpowiedzi do każdego przycisku, które podają nazwę tego koloru po angielsku.

### **Zadanie 3.2**

Do wcześniejszego programu dodaj etykietę LABEL. Ma ona wyświetlać nazwę danego koloru.

### **Zadanie 3.3**

Do wcześniejszego programu dodaj 2 przyciski BUTTON, które blokują i ponownie uaktywniają te zmieniające kolor.

### **Zadanie 3.4**

Do wcześniejszego programu dodaj 2 przyciski BUTTON, które ukrywają i ponownie uaktywniają te zmieniające kolor.