

Instrukcja warunkowa If...Then...ElseIf...Else...End If

Jeśli zaistnieje potrzeba wykonania fragmentu programu pod warunkiem spełnienia pewnego warunku, można skorzystać z instrukcji warunkowej **IF...THEN...END IF**. Instrukcja w najprostszej wersji ma następującą postać:

```
If warunek Then
    instrukcje
End If
```

JEŻELI warunek przyjmuje wartość **PRAWDA**, **WTEDY** wykonaj **instrukcje**.

Bardziej zbudowana postać:

```
If warunek Then
    instrukcje
Else
    else_instrukcje
End If
```

JEŻELI warunek przyjmuje wartość **PRAWDA**, **WTEDY** wykonaj **instrukcje**, **W PRZECIWNYM RAZIE** (jeśli warunek nie przyjmuje wartości **PRAWDA** – czyli przyjmuje wartość **FAŁSZ**), wykonaj **else_instrukcje** będące innym zestawem instrukcji.

Instrukcja IF w najbardziej złożonej postaci:

```
If warunek Then
    instrukcje
ElseIf elseif_warunek Then
    elseif_instrukcje
Else
    else_instrukcje
End If
```

JEŻELI warunek przyjmuje wartość **PRAWDA**, **WTEDY** wykonaj **instrukcje**, **W PRZECIWNYM RAZIE** (jeśli warunek nie przyjmuje wartości **PRAWDA** – czyli przyjmuje wartość **FAŁSZ**) sprawdź **elseif_warunek** i ewentualnie wykonaj **elseif_instrukcje** będące innym zestawem instrukcji, jeżeli żaden z poprzednich warunków nie przyjmował wartości **PRAWDA**, to wykonaj **else_instrukcje**.

Sekcja **ElseIf elseif warunek Then elseif instrukcje** może być powtórzona wielokrotnie w celu sprawdzenia innych przypadków.

Przykład:

```
Dim Liczba, Cyfry As Integer      ' deklaracja zmiennych
Dim Tekst As String
Liczba = 53                       ' wartość początkowa zmiennej
If Liczba < 10 Then
    Cyfry = 1
ElseIf Liczba < 100 Then
    ' jeśli ten warunek jest prawdą, to wykonane będzie następne polecenie.
    Cyfry = 2
Else
    Cyfry = 3
End If

' przypisanie wartości z użyciem postaci jednoliniowej
If Cyfry = 1 Then Tekst = "jedna" Else Tekst = "więcej niż jedna"
```

(przykład pochodzi z [MSDN](#))

Sprawdzenie wartości zmiennej (instrukcja wielokrotnego warunku)

W przypadku, gdy program ma wykonać różne instrukcje w zależności od wartości testowanej zmiennej **zz**, można skorzystać z instrukcji Select Case. Instrukcja ta może mieć postać następującą:

```
Select Case zz
  Case 1 To 5
    ' Wykonaj jeśli  $1 \leq zz \leq 5$ .
  Case 6, 7, 8
    ' Wykonaj jeśli zz jest równe jednej z wartości: 6, 7 lub 8.
  Case 9
    ' Wykonaj jeśli  $zz = 9$ .
  Case Is > 9
    ' Wykonaj jeśli  $zz > 9$ .
  Case Else
    ' Wykonaj w pozostałych przypadkach.
End Select
```

Badana zmienna może być w zasadzie dowolnego typu. Dla typu tekstowego test może wyglądać:

```
Case "czerwony", "zielony"
```

Oczywiście można wykorzystać tylko niektóre konstrukcje, np.:

```
Select Case asd
  Case 1
    Label1.BackColor = Color.Red
    Label1.BackColor = Color.Blue
  Case 2
    Label1.BackColor = Color.Blue
    Label1.BackColor = Color.Red
End Select
```

Instrukcja może być zagnieżdżana, czyli jedna instrukcja Select może znajdować się w drugiej.

Liczby losowe

Może się zdarzyć, że będą potrzebne liczby przypadkowej (losowe). Do tego celu można użyć funkcji `Rnd()`, która generuje liczby mniejsze niż 1 i większe lub równe 0. Można to zapisać następująco:

$$0 \leq \text{liczba_przypadkowa} < 1$$

Jeśli potrzebną będą liczby z innego zakresu, to należy skorzystać ze wzoru:

$$\text{Int}((a - b + 1) * \text{Rnd}() + b)$$

gdzie: `Int()` – funkcja obliczająca część całkowitą z liczby (zmiennoprzecinkowej),

`Rnd()` – funkcja wyznaczająca liczby losowe z przedziału $0 \leq \text{liczba} < 1$,

a – największa liczba z zakresu,

b – najmniejsza liczba z zakresu.

Należy pamiętać, że `Rnd()` jest programem komputerowym i po każdym uruchomieniu programu głównego będzie wyliczać liczby losowe w takiej samej kolejności. Aby temu zapobiec należy skorzystać z funkcji `Randomize()`. Brak parametru w nawiasach spowoduje uzależnienie od zegara systemowego, czyli uzyska się zróżnicowanie generowanych sekwencji liczbowych. Najczęściej wystarczy `Randomize()` wstawić na początek `Sub Form1.Load()`.

Zadanie:

Przypomnij sobie zadanie z lekcji, w którym kolor tła wybranej kontrolki był uzależniony od jedności sekund. Do tego wykorzystana była instrukcja `IF`. Wykonaj podobny program, ale zamiast instrukcji `IF` użyj instrukcji `SELECT` opisanej powyżej (pamiętaj o wykorzystaniu kontrolki `TIMER` oraz dzielenia modulo).

Szczególną uwagę należy zwrócić na postać informacji zwracanej przez polecenie `TimeOfDay`, np. **12:34:46**.