

## Instrukcja warunkowa IF

Wykonuje instrukcję lub blok instrukcji, gdy spełniony jest jakiś warunek logiczny lub też wykonuje inną instrukcję lub blok instrukcji, gdy warunek nie jest spełniony. W pokazanym przykładzie w ramce warunek logiczny - Range("A1") = 0 jest prawdziwy (wartość **TRUE**), gdy w komórce A1 jest zero (lub gdy jest pusta) i wtedy do komórki A2 zostanie wstawiony napis „wartość wynosi zero”. Gdy w komórce A1 znajduje się dowolna liczba (lub tekst), warunek logiczny jest fałszywy (wartość **FALSE**) i do

komórki A2 zostanie wstawiony tekst „wartość jest różna od zera”.

Kolejne dwa przykłady (bez części **ELSE**) – gdy warunek logiczny jest fałszywy, to komórka A2 nie zostanie zmieniona. Jeżeli ma być wykonana tylko jedna instrukcja, można wpisać ją w jednym wierszu.

Jeżeli natomiast wykonanych będzie kilka instrukcji (blok) - ostatnia ramka, wtedy niezbędne jest zakończenie instrukcji warunkowej słowem **END IF**.

```
If Range("A1") = 0 Then
    Range("A2") = "zero"
Else
    Range("A2") = "różne od zera"
End If
```

```
If Range("A1") = 0 Then
    Range("A2") = "zero"
End If
```

```
If Range("A1") = 0 Then Range("A2") = "wartość wynosi zero"
```

```
If Range("A1") > 0 Then
    a = Range("A1")
    b = Range("A2")
    c = Range("A3")
    d = b^2-4*a*c
    Range("A4") = d
End If
```

## Operatory porównania i logiczne

Pełnią identyczną rolę jak w innych językach programowania i w formułach arkusza kalkulacyjnego. Dodatkowe nawiasy podczas stosowania złożonych operacji logicznych nie są konieczne.

### Zmienne, stałe i tablice

W języku VB nie jest konieczne jawne deklarowanie typów zmiennych i stałych, choć podczas pisania dużych projektów deklarując zmienne można zredukować niebezpieczeństwo pojawienia się błędów wynikających z konfliktów nazw i pomyłek w pisowni.

Deklarować można w dowolnym miejscu, w zależności od tego, gdzie chcemy, aby zmienna czy stała była dostępna. Jeśli w procedurze, to tylko procedura ją „widzi”. Jeśli poza procedurą (najczęściej na początku modułu), to można używać ich we wszystkich procedurach modułu.

#### • ZMIENNE

Dim a, b, c - trzy zmienne, ich typ zostanie określony po wpisaniu do nich wartości  
Dim x As Integer - można wpisać liczby całkowite

#### • STAŁE

Const y As Integer = 100 - stała, której przypisano liczbę 100

#### • TABLICE

Dim DniTygodnia(6) - indeksowanie od 0 do 6, dowolne wartości

Dim DniTygodnia(1 To 7) - indeksowanie od 1 do 7, dowolne wartości

Dim DniTygodnia(1 To 7) As String - tylko teksty

Dim Oceny(1 To 35, 1 To 10) As Byte - tablica 10x10 zawierająca liczby

Przypisywanie i odczytywanie wartości do i z elementów tablic odbywa się bardzo podobnie, jak do zmiennych, należy podać tylko indeks – do którego elementu tablicy przypisujemy dane.

```
DniTygodnia(1)="Poniedziałek"
d3=DniTygodnia(3)
Oceny(12,1)=6
Oceny(12,2)=InputBox("Podaj ocenę")
Range("B3")=Oceny(12,3)
Cells(12,4)=Oceny(12,4)
```

operator	opis	przykład (x=10, komórka A1=5)	wynik
<	mniejsze niż	x < 5	False
<=	mniejsze niż lub równe	x <= Range("A1")	False
>	większe niż	10 > Range("A1")	True
>=	większe niż lub równe	10 >= 5	True
=	równe	x = Range("A1") - 5	True
<>	różne (nierówne)	x <> Range("A1") - x	False
And	iloczyn logiczny	10 > 5 And Range("A1") + x = 15 Range("A1") > x And 10 + 5 = 16 (10 < 5) And (Range("A1") + x = 15) (Range("A1")) < (x And 10 + 5 = 16)	True False False False
Or	suma logiczna	(10 > 5) Or (Range("A1") + 5 = 15) (Range("A1") > x) Or (10 + 5 = 16) 10 < 5 Or Range("A1") + 5 = 15 Range("A1") < x Or 10 + 5 = 16	True True True False
Not	zaprzeczenie (negacja)	Not (x = Range("A1") - 5) Not ((10 > 5) Or (Range("A1") + 5 = 15))	False False

## Instrukcja wyboru SELECT CASE

Zamiast stosować złożone w swej konstrukcji zagnieżdżone instrukcje warunkowe można korzystać z instrukcji wyboru. Zaletą jej stosowania jest także to, że ocena wyrażenia wykonywana jest tylko jeden raz i program od razu „skacze” do odpowiedniego bloku, który ma być wykonany. Część zaznaczona na czerwono nie jest obowiązkowa i powinna być zawsze na końcu.

```
d = InputBox("podaj datę")
d = CDate(d)
Select Case Weekday(d)
Case 1
    t = "niedziela"
Case 2
    t = "poniedziałek"
Case 3
    t = "wtorek"
Case 4
    t = "środa"
Case 5
    t = "czwartek"
Case 6
    t = "piątek"
Case 7
    t = "sobota"
Case Else
    t = "nie wpisano daty"
End Select
```

## Ćwiczenia

1. Wczytujemy za pomocą InputBox lub z komórek arkusza trzy liczby do zmiennych a,b,c – są to współczynniki równania kwadratowego. Wyliczamy deltę. Jeżeli delta jest mniejsza od zera to wypisujemy komunikat: „brak pierwiastków” w przeciwnym razie, wyliczamy pierwiastki i wypisujemy w komunikacie.
2. Wczytujemy za pomocą InputBox imię i nazwisko. Program wyświetla komunikat: „Jesteś kobietą” lub „Jesteś facetem” (odczytaj ostatnią literę imienia).
3. Wczytujemy za pomocą InputBox numer studenta (zadanie w oparciu o arkusz STUDENCI). Za pomocą funkcji arkusza WYSZUKAJ.PIONOWO wyszukujemy imię i nazwisko studenta i wypisujemy w okienku MsgBox.
4. Instrukcje języka VB **Selection.Row** i **Selection.Column** podają numer wiersza i kolumny komórki, która jest zaznaczona w bieżącym arkuszu. Na arkuszu STUDENCI zaznaczamy dowolną komórkę arkusza i po wciśnięciu przycisku STUDENT (wstawiamy go na arkusz STUDENCI) procedura podaje w okienku MsgBox imię i nazwisko studenta z wiersza, w którym zaznaczono komórkę.
5. Instrukcje języka VB **Range(„A1”).Select** i **Cells(1,1).Select** powodują wybranie określonej komórki (w tym przypadku komórki A1), a instrukcja **Sheets(„OBSŁUGA”).Select** wybranie arkusza „OBSŁUGA”. Napisz procedurę, w której podajemy za pomocą okienek InputBox numer wiersza, numer kolumny i nazwę arkusza. Program wybiera określony arkusz i zaznacza w nim komórkę.
6. Na arkuszu OBSŁUGA utwórz przycisk skoku do arkusza STUDENCI. Na arkuszu STUDENCI przycisk skoku do arkusza OBSŁUGA
7. Utwórz przycisk na arkuszu STUDENCI, który pokoloruje cały wiersz z zaznaczoną komórką. Utwórz przycisk, który przywróci naturalne kolory wybranemu wierszowi. Użyj metody nagrywania makr. Przyciski KOLORUJ i BEZ KOLORU
8. Za pomocą okienka InputBox wczytujemy numer studenta. Korzystając z funkcji arkusza PODAJ.POZYCJĘ zaznaczamy cały wiersz, ustawiamy kursor na tym numerze i w okienku MsgBox wypisujemy imię i nazwisko studenta. Przycisk SZUKAJ.
9. Miesięczne stypendium dla studenta obliczane jest według następującego algorytmu:
  - obliczamy iloraz punktów przez dochód i mnożony przez 10000
  - jeżeli student jest zamiejscowy, tzn. spoza Rzeszowa otrzymuje dodatek wysokości 100 zł
  - otrzymaną kwotę zaokrąglamy matematycznie do pełnych setek
  - stypendium otrzymują tylko studenci z oceną za egzaminy większą niż lub równą 4,0Zaznaczamy dowolną komórkę w tabeli STUDENCI. Zaznaczamy cały wiersz studenta i w okienku MsgBox wypisujemy wielkość miesięcznego stypendium. Przycisk STYPENDIUM.
10. W nowej kolumnie tabeli STUDENCI tworzymy formułę, która połączy nazwisko i imię studenta (np. kolumna V =**D2&" "&C2**). W górnej części tabeli tworzymy rozwijaną listę z nazwiskami (pasek narzędzi FORMULARZE – POLE KOMBI). We właściwościach tego formantu ustawiamy zakres wejściowy (**\$V\$2:\$V\$374**) i łącze komórki (**OBSŁUGA!\$B\$16**). Tworzymy procedurę która ma działać w identyczny sposób jak w ćwiczeniu 9 – podaje stypendium wybranego studenta. Procedurę przypisujemy do pola kombi zaznaczając je prawym przyciskiem i wybierając **Przypisz makro**. Posortuj tabelę według tej nowej kolumny.

Przykładowe rozwiązania zadań w arkuszu *arkusz – makropolecenia (03).xls*