

# KURS VBScript DLA POCZĄTKUJĄCYCH

Visual Basic Script jest językiem skryptowym, dzięki któremu możesz pisać, np. kod Active Server Pages, skrypty działające w przeglądarce IE lub skrypty dla Windows Script Host. Język ten jest podobny składnią do Visual Basic'a.

## I. KOMENTARZE I PIERWSZY PROGRAM.

Komentarze:

```
' to jest komentarz
// to też jest komentarz
rem i to jest komentarz
```

### **Funkcja MsgBox:**

Instrukcja ta wyświetla dany komunikat w okienku.

```
MsgBox "text",64,"tytuł"
```

Pierwsze pole to jakiś text, drugie to ikonka, a trzecie to tytuł (nagłówek). Dwa ostatnie pola nie są konieczne, natomiast pierwsze musi być.

Przedstawie tutaj podstawowe ikonki jakie możesz zastosować:

- 0 - OK
- 1 - OK Anuluj
- 2 - Przerwij Ponów Próbe Zignoruj
- 3 - Tak Nie Anuluj
- 4 - Tak Nie
- 5 - Ponów Próbe Anuluj
- 16 - Błąd krytyczny
- 32 - Pytanie
- 48 - Ostrzeżenie
- 64 - Informacja

A oto inny sposób na wyświetlenie textu:

```
Dim komunikat 'deklaracja zmiennej
Set komunikat = WScript.CreateObject("WScript.Shell")
komunikat.popup "text"
```

## II. ZMIENNE, STALE.

Zmienne inaczej literały lub stringi, które występują jako textowe lub literowe.

Stosowanie zmiennych jest bardzo przydatne dla programisty.

Zasadę działania zmiennych przedstawię na przykładzie, bo tak oczywiście najlepiej zrozumieć.

Można również użyć dyrektywy Option Explicit, która wymusza deklarowanie zmiennych.

Deklaracja zmiennych:

Set zmienna = obiekt

Dim ..., ..., ...

Przykład 1:

Z1 = "To"

Z2 = " jest"

Z3 = " przykład."

Z = Z1+Z2+Z3

MsgBox Z,64

Przykład 2:

V1 = "To "

I2 = " jest"

R3 = " zastosowanie"

I4 = " zmiennych."

VIRI = V1+I2+R3+I4

MsgBox VIRI,64

Zobacz text VIRI tak samo jak text Z nie zostały wzięte w cudzysłów, ponieważ są to zmienne, w przeciwnym wypadku został by wyrzucony komunikat VIRI lub Z.

Hmmmm, więcej chyba nic nie trzeba tłumaczyć, chyba wszystko wiadomo ;>, zresztą wszystko widać w kodzie.

A tutaj przedstawię ciekawszy przykład, z zastosowaniem kodu ASCII.

No to kodzimy ;>...

Przykład 3:

Dim mychar, ascii

Set WshShell = Wscript.CreateObject("Wscript.Shell")

k1 = chr(107)

k2 = chr(111)

k3 = chr(100)

k4 = chr(32)

k5 = chr(65)

k6 = chr(83)

k7 = chr(67)

k8 = chr(73)

k9 = chr(73)

k = k1+k2+k3+k4+k5+k6+k7+k8+k9

komunikat = k

t1 = chr(65)

t2 = chr(83)

t3 = chr(67)

t4 = chr(73)

```
t5 = chr(73)
t = t1+t2+t3+t4+t5
tytuł = t
Set WshShell = Wscript.CreateObject("Wscript.Shell")
ascii = WshShell.Popup(komunikat,5,tytuł,29)
rem okienko z komunikatem będzie widoczne na ekranie przez
rem 5 sekund po czym samo zniknie.
```

Lub też tak:

```
Dim mychar, ascii
Set WshShell = Wscript.CreateObject("Wscript.Shell")
k = Chr(65) & Chr(83) & Chr(67) & Chr(73) & Chr(73)
komunikat = k
ascii = WshShell.Popup(komunikat)
```

Definiowanie stałej: Const stała = wartość

### **III. FUNKCJA LEN, FUNKCJA INPUTBOX ORAZ INSTRUKCJA WARUNKOWA.**

Funkcja Len:

Funkcja ta służy do określania długości zmiennych (ile liter ma dany wyraz czy zdanie).

Przykład 1:

```
Msgbox Len("Zastosowanie funkcji Len")
```

Funkcja InputBox:

Polega na pobieraniu wartości (z klawiatury) od użytkownika.

Przykład 1:

```
V = InputBox("Wprowadź liczbę: ")
MsgBox "Wprowadzona liczba to:" & V
```

Przykład 2:

```
V=InputBox("Podaj swoje imię")
S="tu jest text "+V+" a tutaj jest inny text"
Msgbox S,64
```

W powyższym przykładzie jest również zastosowane łączenie stringów (+V+).

Instrukcja warunkowa:

If warunek Then

instrukcja I

...

Else

...

instrukcja II  
End if

Jeżeli warunek zostanie spełniony to zostanie wykonana instrukcja I, natomiast gdy warunek nie zostanie spełniony to zostanie wykonana instrukcja II.

Przykład 1:

```
V = InputBox("Podaj swój ulubiony język programowania")
If V = "VBScript" Then
MsgBox "Dobry wybor",64
Else
MsgBox "Nie lubisz VBScript",64
End if
```

Przykład 2:

```
Key = CInt(Day(Date))
If Key = 12 Then
MsgBox "tak"
Else
MsgBox "nie"
End if
```

W powyższym przykładzie 12-nastego dnia każdego miesiąca program wyrzuca text "tak", natomiast w pozostałe dni wyrzuca tekst "nie".

#### **IV. PĘTLE.**

Pętle zostały wymyślane po to, aby ułatwić życie programiście ;>. Jeżeli chcesz, aby jakiś tekst został wyświetlony na ekranie 20 razy, możesz napisać 20 MsgBox'ow, heh, ale lepiej użyć do tego pętli, zrobisz to o wiele szybciej i skrócisz o wiele kod programu.

For...Next:

Przykład 1:

```
For n = 1 to 5
MsgBox "To jest pętla"
Next
```

Dany MsgBox zostanie wyświetlony na ekranie 5 razy.

Przykład 2:

```
n = n+1
For n = 1 to 5
MsgBox "To jest pętla" & n
Next
```

Do...Loop:

Przykład 1:

```
Set M = CreateObject("Scripting.FileSystemObject")
Set Dat = M.OpenTextFile("c:\autoexec.bat")
Do
n = n + 1
Wiersz = Dat.ReadLine
Wscript.echo "Wiersz nr " & n & ": " & Wiersz
Loop Until Dat.AtEndOfStream = True
```

Wyświetla całą zawartość pliku autoexec.bat.

Przykład 2:

```
Do Until DefResp = vbNo
MyNum = Int (2 * Rnd + 1)
DefResp = MsgBox (MyNum & " Wybierz numer:", vbYesNo)
Loop
If MyNum = Int(2) Then
MsgBox "Wybrałeś numer 2"
Elseif MyNum = Int(1) Then
MsgBox "Wybrałeś numer 1"
End if
```

Przykład 3:

on error resume next 'jeśli wystąpią błędy wykonuj dalej skrypt

```
Dim fso, MyFile
Do Until MyFile = fso.CreateTextFile("C:\Pętla.bat", True)
Set fso = CreateObject("Scripting.FileSystemObject")
Set MyFile = fso.CreateTextFile("C:\Pętla2.bat", True)
MyFile.WriteLine("@echo off")
MyFile.WriteLine("echo DESIGNED BY v1rhuz5")
MyFile.Close
Loop
```

**For Each...Next:**

Przykład 1:

```
Set MyFiles = CreateObject("Scripting.FileSystemObject")
Set Folder = MyFiles.GetFolder("c:\")
Set WszystkiePliki = Folder.Files
For Each X in WszystkiePliki
L = L & X.Path & chr(9) & X.DateLastAccessed & chr(13)
Next
Wscript.echo L
```

Wyświetla wszystkie pliki z dysku c:\.

Instrukcja Exit For umieszczona w środku pętli powoduje jej zamknięcie (wyjście z pętli).

## V. FUNKCJE.

Funkcje:

Stosowanie funkcji w programie jest bardzo przydatne. Jeżeli w programie jest zawarty jakiś fragment kodu, który chcesz, np. powtórzyć kilkakrotnie to, wystarczy że umieścisz go w funkcji, którą potem możesz wywołać w dowolnym miejscu programu.

Przykład 1:

```
Function text
MsgBox "Jestem w funkcji."
End Function
text
text
```

Przykład 2:

```
MsgBox "Zastosowanie funkcji " & przykład
Function przykład
MsgBox "To jest właśnie funkcja."
End Function
```

Przykład 3:

```
Key = CInt(Month(Date))
If Key = 12 Then
przykład
end if
Function przykład
MsgBox "To jest właśnie funkcja."
End Function
```

Dany MsgBox zostanie wyświetlony w miesiącu grudniu.

Instrukcja Exit Function umieszczona w środku funkcji powoduje jej zamknięcie (wyjście z funkcji).

Przykład 4:

```
Sub test
MsgBox "Przykład funkcji."
End Sub
test()
```

## **VI. TABLICE.**

Tablica jest to można powiedzieć zmienna, tylko taka, która może pomieścić kilka wartości. Poniższy przykład bardzo dobrze pokazuje zastosowanie tablic.

Przykład 1:

```
Dim Tab(5)
Tab(1)="V"
Tab(2)="i"
Tab(3)="R"
Tab(4)="i"
Tab(5)="i"
For n = 1 to 5
MsgBox Tab(n)
Next
```

## **VII. INSTRUKCJA UCASE LCASE.**

Instrukcje te zmieniają zmienne tak, aby składały się one z samych dużych lub małych liter.

Ucase - duże litery

Lcase - małe litery

Przykład 1:

```
V = InputBox("Pisz:", "")
V = Ucase(V)
MsgBox V, 64
V = InputBox("Pisz:", "")
V = Lcase(V)
MsgBox V, 64
```

Przykład 2:

```
podaj = InputBox ("password:")
pass = "VBS"
If ucase(podaj) = pass Then
wynik = MsgBox ("correct")
Else
wynik = MsgBox ("incorrect")
End If
```

## **VIII. INSTRUKCJA MID, LEFT, RIGHT.**

Instrukcja Mid:

Instrukcja ta wyświetla środkowe znaki.

Przykład 1:

```
MsgBox Mid("Instrukcja Mid",2,6)
```

Teraz na ekranie zostanie wyrzucony sam napis: nstruk.

Instrukcja Left:

Instrukcja ta wyświetla podaną ilość pierwszych znaków.

Przykład 2:

```
MsgBox Left("Instrukcja Left",10)
```

Teraz na ekranie zostanie wyrzucony sam napis: Instrukcja.

Instrukcja Right:

Instrukcja ta wyświetla podaną ilość ostatnich znaków.

Przykład 3:

```
Msgbox Right("Instrukcja Right",5)
```

Teraz na ekranie zostanie wyrzucony sam napis: Right.

## **IX. OPERACJE NA PLIKACH I SPECIAL FOLDER.**

Tutaj znajdują się podstawowe operacje na plikach, które możesz użyć w swoim programie.

Usuwa plik z dysku:

```
Dim fso  
Set fso = CreateObject("Scripting.FileSystemObject")  
fso.DeleteFile("c:\plik.txt")
```

Usuwa folderu z dysku:

```
Dim fso  
Set fso = CreateObject("Scripting.FileSystemObject")  
fso.DeleteFolder("c:\folder")
```

Tworzy plik na dysku:

```
Dim fso, MyFile  
Set fso = CreateObject("Scripting.FileSystemObject")  
Set MyFile = fso.CreateTextFile("C:\plik.txt", True)  
...
```

Aby utworzyć dany plik z zawartością, wystarczy dopisać:

```
...  
MyFile.WriteLine("To jest plik.txt z zawartym tekstem")  
MyFile.WriteLine("by v1rhuz5")
```



MyFile.Close

Tworzy folder na dysku:

```
Dim fso, f
Set fso = CreateObject("Scripting.FileSystemObject")
Set f = fso.CreateFolder("c:\folder")
```

Kopiuje plik:

```
Dim fso
Set fso = CreateObject("Scripting.FileSystemObject")
fso.CopyFile "c:\plik.txt", "c:\my documents\"
```

Kopiuje folder:

```
Dim fso
Set fso = CreateObject("Scripting.FileSystemObject")
fso.CopyFolder "c:\plik", "d:\"
```

Specjalne foldery:

Służą one do..., np. jeżeli napiszemy program i chcemy, aby się on przekopiował do katalogu c:\windows, możemy napisać tak:

```
Dim fso
Set fso = CreateObject("Scripting.FileSystemObject")
Set c = fso.GetFile(WScript.ScriptFullName)
c.Copy("C:\windows\jakiśplik.vbs")
```

Teraz gdy damy komuś taki program i soba ta będzie miała katalog C:\win98 to program nasz nie zadziała. I do tego właśnie służą foldery specjalne, lepiej więc napisać tak:

```
Dim fso
Set fso = CreateObject("Scripting.FileSystemObject")
Set c = fso.GetFile(WScript.ScriptFullName)
Set dirwin = fso.GetSpecialFolder(0)
c.Copy(dirwin & "\jakiśplik.vbs")
```

Nasz program teraz napewno zadziała, bez względu na to jaką nazwę będzie miał katalog windoza.

Special folder:

- 0 - Windows folder
- 1 - System folder
- 2 - Temp folder

## X. PROGRAM.

Tutaj znajduje się kod bardzo prostego programu służący jako przykład do nauki podstaw vbs:

```
----- cut -----  
Dim Num, opcja  
MsgBox "Program pochodzi z kursu: KURS VBScript DLA  
POCZ?TKUJ?CYCH",64,"(c)v1rhuz5"  
Num= InputBox("1.Versja VBS 2.Data 3.Użytkownik 4.Inf-Windows 5.Kalkulator")  
  
If IsNumeric(Num) then  
Select Case Num  
Case 1 opcja = jeden  
Case 2 opcja = dwa  
Case 3 opcja = trzy  
Case 4 opcja = cztery  
Case 5 opcja = pięć  
End Select  
End if  
  
Function jeden  
MsgBox wersja, "Versja"  
End function  
  
Function wersja  
Dim L  
L = ""  
L = ScriptEngine & " Version "  
L = L & ScriptEngineMajorVersion & "."  
L = L & ScriptEngineMinorVersion & "."  
L = L & ScriptEngineBuildVersion  
wersja = L  
End Function  
  
Function dwa  
Dim komunikat  
Dim dzisiaj  
set komunikat = WScript.CreateObject("WScript.Shell")  
dzisiaj = Date  
komunikat.popup "Dzisiejsza data to " & dzisiaj, 0,"Bieżąca data"  
End function  
  
Function trzy  
Set objNet = WScript.CreateObject( "WScript.Network" )  
Wscript.Echo objNet.ComputerName & " - " & objNet.UserName  
End function
```

```
Function cztery
Dim fso
Set fso = CreateObject("Scripting.FileSystemObject")
Set Wsh = CreateObject("WScript.Shell")
Wsh.Run("rundll32.exe shell32.dll,ShellAboutA")
End function
```

```
Function piec
Dim fso
Set fso = CreateObject("Scripting.FileSystemObject")
Set Wsh = CreateObject("WScript.Shell")
Wsh.Run("calc.exe")
End function
```

----- cut -----

End