

VB'de Veri Türleri¹

Byte	1	İşaretsiz tamsayı
Integer	2	İşaretili Tamsayı
Long	4	İşaretili Tamsayı
Single	4	Gerçel sayı
Double	8	Gerçel sayı
Currency	8	Gerçel sayı
Decimal	14	Gerçel sayı
Boolean	2	Doğru/Yanlış (0/1)
String	10	Metin
Variant	-	Gevşek veri türü
Date	8	Tarih/Zaman
Object	4	Nesne veri türü
Type	-	Kullanıcı tanımlı

Byte:

C'deki "unsigned char" veri türüne karşılık gelir. 0-255 arasında değerler alabilir.

Integer:

C'deki "int" veri türüne karşılık gelir. -32 768 ile 32767 arasında değerler alır.

Long:

-2 147 483 648 ile 2 147 483 647 arasında değer alır.

VB'de kullanılan bu 3 tür tamsayı niteliğindeki veri türlerine sahip değişkenlere, bir gerçel sayı atanmaya çalışılırsa, en yakın atanacak değer en yakın tam sayıya yuvarlanarak atanır.

Single:

C'deki "float" veri türüne karşılık gelir. A^B şeklindeki gösterimde A ve B integer veri türündedir. Örneğin 3,12 ifadesini 312×10^{-2} şeklinde gösterirsek $A=312$ ve $B=-2$ şeklinde toplam (2 integer) 4 byte genişliğinde bellekte yer tutar.

Double:

A^B şeklindeki gösterimde A ve B long veri türüne sahiptirler. Bu nedenle bellekte toplam 8 byte genişliğinde yer tutarlar.

Currency:

Gerçel sayı olmasına karşılık ondalık kısmı en fazla 4 dijittir. Ondalık kısmı 4 dijitten fazla olan kesim yuvarlanarak 4 dijite indirgenir. Daha çok tamsayısal kısmın büyük değerler aldığı, ondalık bölümlerin ise önemsiz olduğu parasal verilerde kullanılır.

Dim X As Currency

X=1.2345678 ise X=1.2346 şeklinde atanır.

Decimal:

¹ Bilgilerin bir kısmı için, Microsoft Visual Basic Books Online e-kitabından yararlanılmıştır.

Currency gibidir, ancak currency'nin aksine 29 dijite kadar ondalık bölümü tutabilir. Hassas matematiksel verilerde kullanılır. Ancak en önemli özelliği, bir değişkenin veri türü olarak tanımlanamaz. Bunun yerine variant veri türüne sahip bir değişkene CDec(variant_degisken) şeklindeki tür dönüşümleri ile kullanılır.

Boolean:

C'de bulunmayan bir veri türüdür. Yalnızca 0 ya da 1 değeri alabilir. 0 değeri yanlış (false) ve 1 değeri doğru (true) şeklinde değerlendirilir. VB'de 0/1 değerleri yerine True/False şeklinde de atama yapılabilir. Bilindiği gibi bu tür değerler aynı zamanda, mantıksal işlem sonuçlarıdır.

String:

C'deki karakter dizilerine karşılık gelir. Yaklaşık 2 milyar karakteri kadar atama yapılır. C'deki "char *degisken" sözdiziliminde olduğu gibi, değişken uzunlukta bir yapıya sahiptir. Atama esnasında kendini boyutlandırır.

C'deki "char degisken[50]" sözdiziliminde olduğu gibi değişken uzunluğun aksine sabit uzunluklu tanımlamada yapılabilir.

Dim X As String	'(değişken uzunluk)
X="Hacettepe Üniversitesi"	'(Atama esnasında boyutlandırılır)
Dim X As String*50	'(sabit uzunluk)
X="Hacettepe Üniversitesi"	'(En fazla 50 karakter atanabilir)

Bu durumda, Dim X As String*1 şeklinde yapılan bir tanımlama C'deki "char" veri türüne karşılık gelir.

Variant:

Yeni nesil programlama dillerinde, bir gevşek veri türü kullanılmaktadır. VB'de bu veri türü variant'tır. Bu veri türünde belirli bir veri söz konusu değildir. Variant veri türüne değişkenler, atanan değere göre kendisini şekillendirebilir. Örneğin;

Dim X As Variant	
X=12	'Tamsayı türüne dönüştü
X=65536	'Taşması hatası oluşmayacak, çünkü değişken Long veri türüne dönüştü
X=6.65536	'Single veri türüne dönüştü
X="BÖTE"	'String veri türüne dönüştü

(Tanımlanmayan değişkenler Variant veri türüne sahiptir.)

Değişkenlere ilişkin veri türleri, çalışma anında kendiliğinden değişebilir ya da veri dönüşüm fonksiyonları ile de dönüştürülebilir. Bu fonksiyonlar aşağıda verilmiştir.

Fonksiyon	Hedef Veri Türü
CBool	Boolean
CByte	Byte
CCur	Currency
CDate	Date
Cdbl	Double
CInt	Integer
CLng	Long
CSng	Single
CStr	String
CVar	Variant
CVErr	Error

Değişkenlere ilişkin diğer fonksiyonlar:

IsEmpty(degisken):

Belirtilen değişkene daha önceden değer atanmamış ise geriye True değeri dönderir.

Dim Degisken	
Dim Kontrol	
Kontrol = IsEmpty(Degisken)	'Dönen değer: True.
Degisken = Null	'Null atandı, "\0" değeri var.
Kontrol = IsEmpty(Degisken)	'Dönen değer: False.
Degisken = Empty	'Empty atandı,
Kontrol = IsEmpty(Degisken)	'Dönen değer: True.

Not: Empty, hiç değer atanmamış anlamındadır.

IsNumeric(degisken):

Değişkenin içeriğinde sayısal bir değer var ise; True, değilse False değeri dönderir.

Do	Sayi = InputBox("Bir Sayi Giriniz")
Loop Until	IsNumeric(Sayi)
	MsgBox "KareKökü: " & Sqr(Sayi)

Dim Degisken, Kontrol	
Degisken = "53"	
Kontrol = IsNumeric(Degisken)	' True.
Degisken = "459.95"	
Kontrol = IsNumeric(Degisken)	' True.
Degisken = "45xyz"	
Kontrol = IsNumeric(Degisken)	' False.

isArray(değişken):

Değişkenin, bir dizi olup olmadığını belirler. Eğer değişken bir dizi ise True, değilse False değeri döndürür.

Diziler

Diziler boyutlu değişkenler olarak ifade edilebilir. Bu nedenle tekil değişkenlerin aksine, boyutları da belirtilmelidir.

Dim X(5) As Integer 'Tek boyutlu dizi

Dim X(1 To 5) As Integer

Dim X(3,4) As Integer 'İki boyutlu dizi

Dim X(3,4,5) As Integer 'Üç boyutlu dizi

Not: Boyut sınırları, tanımlama esnasında belirlenen diziler "sabit diziler" olarak adlandırılırlar. Boyut sınırları tanımlama esnasında belirlenmeyen diziler "dinamik dizi" olarak adlandırılır ve bu tür diziler çalışma anında boyutlandırılabilir özelliği taşırlar. Sabit diziler, çalışma anında boyutlandırılmazlar.

Dim X() As Integer 'Dinamik dizi

ReDim X(5) As Integer 'Çalışma anında yeniden boyutlandırıldı.

Redim Preserve X(3) 'Çalışma anında yeniden boyutlandırıldı.

Dinamik Diziler:

Tanımlama esnasında boyut bildirme zorunluluğu olmayan ancak program içerisinde defalarca boyutlandırılabilir yeteneği olan bir dizi türüdür. Boyutlandırma esnasında;

Redim

Redim Preserve

Anahtar kelimeleri (Keywords) kullanılır. Aralarındaki fark ise, "Redim" ile yapılan boyutlandırma da dizinin içerisine daha önce bir değer atanmış ise hepsi silinir. "Redim Preserve" ile yapılan boyutlandırmalarda ise dizi içerisinde daha önce değer atanmış ise eski değerler korunur.

"Redim Preserve" ile yapılan boyutlandırmaların en önemli dezavantajı ise, yalnızca kullanılan son boyutta değişiklik yapılabilmesi.

Redim Preserve X(3)

Redim Preserve X(3,4)

Redim Preserve X(3,4,5)

Yukarıdaki örneklerde “Redim Preserve” ile yalnızca koyu işaretli boyutlar yeniden boyutlandırılabilir.

Dim Dizi() As Integer	'Dizi tanımlaması
Redim Dizi(5)	'Yeniden boyutlandırma
For I = 1 To 5	'Döngü: 5 kere dön
Dizi(I) = I	'Diziye atama yapılması.
Next I	

Dizilere ilişkin *Erase* fonksiyonu:

Erase DiziAdı

Bu fonksiyon, belirtilen dizi;

Dinamik ise	→	Diziyi bellekten atar.
Sayısal ise	→	Tüm elemanlarını 0 yapar.
String ise	→	Tüm elemanlarını "" yapar.

UBound ve LBound Fonksiyonları:

Dizinin belirtilen boyutunun üst sınırını (UBound) ve alt sınırını (LBound) geri dönderirler.

Dim A(1 To 100, 0 To 3, -3 To 4)

UBound(A, 1)	'100 değeri dönderir
UBound(A, 2)	'3 değeri dönderir
UBound(A, 3)	'4 değeri dönderir
LBound(A, 1)	'1 değeri dönderir
LBound(A, 2)	'0 değeri dönderir
LBound(A, 3)	'-3 değeri dönderir

IIF Koşulu

IIF(koşul, doğru,yanlış)

iif(x>0, MsgBox "Pozitif", MsgBox "Negatif")

IF-ELSEIF-ELSE

Dim Sayı, Basamak, ifade

Sayı = 53

If Sayı < 10 Then

 Basamak = 1

ElseIf Number < 100 Then

 Basamak = 2

Else

 Basamak = 3

End If

If Basamak = 1 Then ifade = "Bir" Else ifade = "Birden Fazla"

SELECT CASE

```
Select Case ifade
[Case durumlar-n
    [deyimler-n]] ...
[Case Else
    [aksidurumlar]]
End Select
```

```
Dim Sayi
Sayi = 8.
Select Case Sayi
Case 1 To 5
    MsgBox "Sayı 1- 5 arasında"
Case 6, 7, 8
    MsgBox "Sayı 6- 8 arasında"
Case Is > 8 And Number < 11
    MsgBox "Sayı 9 ya da 10'dur"
Case Else
    MsgBox "Sayı 1- 10 arasında değil"
End Select
```

DO-LOOP Döngüleri

```
Dim Kontrol, Sayi
Kontrol = True: Sayi = 0
Do
    Do While Sayi < 20
        Sayi = Sayi + 1
        If Sayi = 10 Then
            Kontrol = False
            Exit Do
        End If
    Loop
Loop Until Kontrol = False
```

'Dış Döngü
'İç Döngü
'Sayaç artımı
'Eğer koşul doğru ise
'Kontrol değerini false yap
'İç döngüden çık (koşula bağlı)

'Dış döngü bitti