

Változók

Készítette: Vastag Attila

2016

Amikor programot írunk, akkor szükség lehet tárolókra, ahová az adatainkat ideiglenesen eltároljuk. Ezeket a tárolókat változóknak nevezzük.

A változók a memória egy (vagy több) cellájára hivatkozó leírók. Egy változót a következő módon hozhatunk létre C# nyelven:

típus *változónév*;

A változónév első karaktere csak betű vagy alulvonás jel () lehet, a többi karakter szám is. Lehetőleg kerüljük az ékezetes karakterek használatát.

Konvenció szerint a változónevek kisbetűvel kezdődnek. Amennyiben a változónév több szóból áll, akkor célszerű azokat a szóhatárnál nagybetűvel “elválasztani” (pl. pirosAlma, vanSapkaRajta, stb).

Típusok

C# típus	.NET típus	Méret (byte)	Leírás
byte	<i>System.Byte</i>	1	Előjel nélküli 8 bites egész szám (0..255)
char	<i>System.Char</i>	2	Egy Unicode karakter
bool	<i>System.Boolean</i>	1	Logikai típus, értéke igaz(1 vagy true) vagy hamis(0 vagy false)
sbyte	<i>System.SByte</i>	1	Előjeles, 8 bites egész szám (-128..127)
short	<i>System.Int16</i>	2	Előjeles, 16 bites egész szám (-32768..32767)
ushort	<i>System.UInt16</i>	2	Előjel nélküli, 16 bites egész szám (0..65535)
int	<i>System.Int32</i>	4	Előjeles, 32 bites egész szám (-2147483648.. 2147483647).
uint	<i>System.UInt32</i>	4	Előjel nélküli, 32 bites egész szám (0..4294967295)
float	<i>System.Single</i>	4	Egyszeres pontosságú lebegőpontos szám
double	<i>System.Double</i>	8	Kétszeres pontosságú lebegőpontos szám
decimal	<i>System.Decimal</i>	16	Fix pontosságú 28+1 jegyű szám
long	<i>System.Int64</i>	8	Előjeles, 64 bites egész szám
ulong	<i>System.UInt64</i>	8	Előjel nélküli, 64 bites egész szám
string	<i>System.String</i>	N/A	Unicode karakterek szekvenciája
object	<i>System.Object</i>	N/A	Minden más típus őse

Egy karakter tarolására szolgál

Ha logikai értéket definiálunk

Ha egész számot definiálunk

Ha lebegő pontos számot definiálunk

Ha tizedes számot definiálunk

Ha karakterek sorozatát tárolnánk

_____ szám = 452;

_____ hanyados = 34.7975;

_____ bBetu = 'b';

_____ igaz = false ;

_____ hamis = false;

_____ a = 97;

_____ vezetekNev = "Virag";

Deklaráció és definíció

Egy változó (illetve lényegében minden objektum) életciklusában megkülönböztetünk deklarációt és definíciót. A deklarációnak tartalmaznia kell a típust és azonosítót (ilyenkor a c# ha numerikus adatról van szó, érténként 0-t rendel hozzá, még szöveg esetén *null* értéket), a definícióban pedig megadjuk az objektum értékét. Értelemszerűen a deklaráció és a definíció egyszerre is megtörténhet.

```
int x;           // deklaráció
x = 10;          // definíció
int y = 11;      // deklaráció és definíció
```

Deklaráció és definíció

Ha egy változót csak deklarátunk, akkor azoknak a c# fordító egy alapértelmezett értéket rendel hozzá.

Változó	Alapértelmezett alapérték	Mit kell nekünk megadni
int	0	
double	0.0	
char	'\0'	
string	Null	string.Empty
Bool	false	

Típusok

A C# erősen (statikusan) típusos nyelv, ami azt jelenti, hogy minden egyes változó típusának ismertnek kell lennie fordítási időben, ezzel biztosítva azt, hogy a program pontosan csak olyan műveletet hajthat végre amire valóban képes. A típus határozza meg, hogy egy változó milyen értékeket tartalmazhat, illetve mekkora helyet foglal a memóriában.

A következő táblázat a C# beépített típusait tartalmazza, mellettük ott a .NET megfelelőjük, a méretük és egy rövid leírás:

HÁZI FELADAT

int szam = _____;

_____ hanyados = 34.7975;

char bBetu = _____;

bool esik = _____;

_____ vizes = false;

double eredmeny = _____;

_____ tanuloSzama = 16;

_____ vezetekNev = "Hapci";

string keresztNev = _____;

Házi feladat:

Hozza létre a megfelelő változókat és tárolja el benne az adatokat az alábbi röplabdázóról:

Nev: **Katarzyna Skowronska-Dolata**

Nemzetiség: **Lengyel**

Születési év: **1983**

Születési hely: **Warszawa**

Magasság: **1.89**

Tömeg: **75**

Ütő magassága: **317**

Sánc magassága: **302**

Mez száma: **2**

Csapatkapitány: **igen**

Kapitany jele: **C**