

KIMENET

az adatok közzlése a felhasználóval

Készítette: Vastag Attila

2016

Egy programnak csak akkor van értelme, ha a feldolgozás után az valamilyen adatok közöl a felhasználóval, akinek ez információt jelent. Az adat a felhasználóval való közlés alapértelmezett megjelenítési eszköze a monitor, vagyis az adatok megjelenítése a képernyőn.

Konzol alkalmazásoknál (ilyeneket írunk mi) ez a következő utasítással lehetséges:

```
Console.Write("ide irom a kiírandó szöveget");  
Console.WriteLine("ide irom a kiírandó szöveget");
```

A két utasítás közt az a különbség, hogy az első után a gép nem ugrik új sorba, még a második után igen.

A valós életben nagyon ritkán fordul elő, hogy konstans adatot kell megjeleníteni a képernyőn a felhasználónak, hisz az adatok feldolgozása után, soha nem tudjuk mit is kellene a felhasználó felé küldeni, csak ha vége az adatfeldolgozó folyamatnak. Ilyenkor az eredmény(eke)t kell közölni a felhasználóval.

```
string nev = "Vastag Atila";  
int eletkor = 38;  
double atlag = 9.73;
```

```
Console.WriteLine("A nevem: {0}, eletkorom: {1}, tanulmanyi atlagom : {2}!", nev, eletkor, atlag);
```



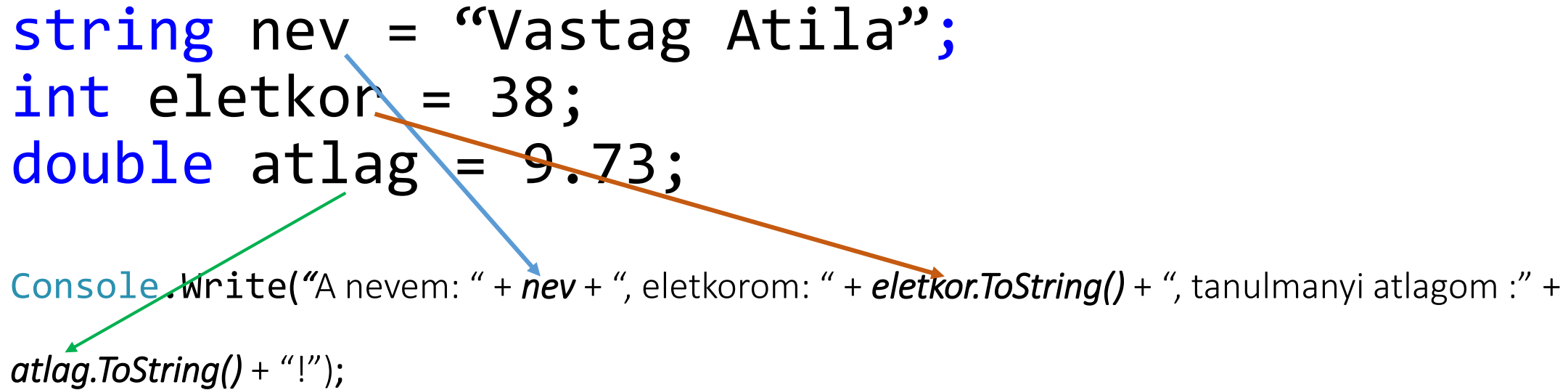
A képernyőn a következő szöveg jelenik meg:

A nevem: Vastag Atila, eletkorom: 38, tanulmanyi atlagom: 9.73!

Az előző kiírási mód, amely egyben az ajánlott is egyben, létezik egy másik módja is, *string* típusú változók összefűzésével:

```
string nev = "Vastag Atila";  
int életkor = 38;  
double atlag = 9.73;
```

```
Console.WriteLine("A nevem: " + nev + ", életkorom: " + életkor.ToString() + ", tanulmányi átlagom : " +  
atlag.ToString() + "!");
```



A képernyőn a következő szöveg jelenik meg:

A nevem: **Vastag Atila**, életkorom: **38**, tanulmányi átlagom: **9.73**!

A *string* típusú adatokat úgy is mondhatnánk, hogy „összeadtuk”, azaz mindig a meglévő karakter sorozat végéhez adtuk hozzá az újat.

Hátránya az, hogy ha nem *string* típusú változót akarunk kiírni, akkor az először a *string* típusra kell átkonvertálni, erre szolgál a **ToString()** függvény (funkciója: alakítsd át *string*-é), amelyet minden nem *string* típusú változón meg kell hívni ha ezt a kiírási módot választanánk.

MS Visual Studio 2015-től, azaz a c# 6.0 verziójától a következő képen is formattálhatunk meg *string*-et kiírásra.

```
string nev = "Vastag Atila";  
int életkor = 38;  
double atlag = 9.73;
```

```
Console.Write($"A nevem: {nev}, életkorom: {életkor}, tanulmányi atlagom : {atlag}!");
```



A képernyőn a következő szöveg jelenik meg:

A nevem: Vastag Atila, életkorom: 38, tanulmányi atlagom: 9.73!

A kiírásnál használhatunk speciális karaktereket is.

\n – új sor === Environment.NewLine

```
Console.WriteLine(Environment.NewLine);
```

```
Console.WriteLine("A következő mondat új sorba fog kerülni.\nA második mondat új sorban!");
```

kimenet

A következő mondat új sorba fog kerülni.

A második mondat új sorban!

\t – tabulátor

```
Console.WriteLine("0\t1\t2\t3\t4\t5\t6\t7");
```

kimenet

0 1 2 3 4 5 6 7

1 – Hozzunk létre megfelelő változó típusokat a következő értékekkel:**Torino Cobra SportsRoof, 1970.** Majd a képernyőre írjuk ki:

*A Ford autógyár **1970**-ben gyártotta le a **Torino Cobra SportsRoof** modellt.*

2 – Hozzunk létre megfelelő változó típusokat a következő értékekkel:**i7, 8700, 4.6:**

*Az INTEL **i7** sorozatjelzésű mikroprocesszor **8700** típusának munkafrekvenciája **4.6GHz**.*

3 – Hozzunk létre megfelelő változó típusokat a következő értékekkel:**1644, 1368, 7200, Kínai Nagyfal, Ming:**

*A **Ming** dinasztia **1368** és **1644** közt építette a ma látható **Kínai Nagyfal** **7200km** hosszú szakaszát.*

4 – Hozzunk létre megfelelő változó típusokat a következő értékekkel:**Sukhoi, SU-35S, 2008, 2.4, m/s:**

*Oroszország legmodernebb vadászrepülőgépe a **2008**-ban bemutatott **Sukhoj** gyártotta **SU-35S** amely képes a **2.4 Mach** sebességre, ami **720m/s**-nak felel meg.*