

# Függvények

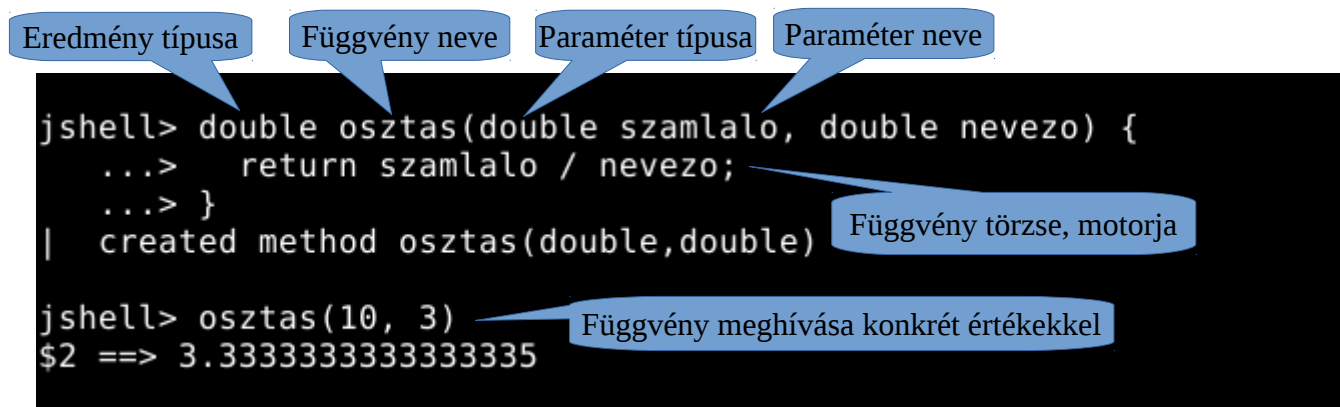
A számítógépes programozásban a függvények összetett műveletek, logikák, algoritmusok tárolására szolgálnak. Hasonlítanak a matematikából is ismert függvényekre. Lehet néhány bemeneti adatuk (ezeket paramétereknek nevezzük), és lehet visszatérési értékük.

Matematikai függvény például:  $f(x) = x^2 - 2x + 1$

Ha ennek a függvényke konkrét értéket adunk be (x-nek egy számot helyettesítünk be), kapunk egy eredményt. Pl.  $f(0) = 1$ ,  $f(1) = 0$ ,  $f(2) = 1$ ,  $f(3) = 4$  ...

Mivel a Java-ban minden adatnak (változónak) típusa van, a függvények esetén is meg kell határoznunk a bemeneti és kimeneti adatok típusát.

Példa:



```
jshell> double osztas(double szamlalo, double nevező) {  
    ...> return szamlalo / nevező;  
    ...> }  
| created method osztas(double,double)  
  
jshell> osztas(10, 3)  
$2 ==> 3.3333333333333335
```

Callout boxes:

- Eredmény típusa (points to 'double' in the return statement)
- Függvény neve (points to 'osztas')
- Paraméter típusa (points to 'double' in the parameter list)
- Paraméter neve (points to 'szamlalo' and 'nevező')
- Függvény törzse, motorja (points to the function body)
- Függvény meghívása konkrét értékekkel (points to the function call)

A függvényeket természetesen változókkal is meg lehet hívni, ill. a függvény visszatérési értékét megfelelő típusú változónak átadni.

A fenti példa folytatása:

```
jshell> double x = 7; double y = 11  
x ==> 7.0  
y ==> 11.0  
  
jshell> double z = osztas(x, y)  
z ==> 0.6363636363636364
```

A tiszta függvények csak a bemeneti paraméterek adatainak felhasználásával állítják elő az eredményt (a fenti példa ilyen). A függvények azonban „látják” a környezetük változóit is.

Példa:

```
jshell> double kamatlab = 7.3
kamatlab ==> 7.3

jshell> double kamat(int toke) {
...>     return toke * kamatlab / 100;
...> }
| created method kamat(int)

jshell> kamat(12000)
$3 ==> 876.0

jshell> kamat(895578)
$4 ==> 65377.193999999996
```

A fenti példában a kamatlab környezeti változóként van definiálva. A kamatot számoló függvény bemeneti paraméterként nem kapja meg a kamatlábat, csak a tőkét. A kamatlábat közvetlenül éri el. Ez természetesen azt is jelenti, hogy a kamatlab változása esetén a függvény működése is változik.

```
jshell> kamatlab = 8
kamatlab ==> 8.0

jshell> kamat(100)
$6 ==> 8.0

jshell> kamatlab = 9.7
kamatlab ==> 9.7

jshell> kamat(100)
$8 ==> 9.7
```

Továbbá fontos az is, hogy a függvények nem csupán „látják” a környezetük változóit, hanem képesek azok értékét módosítani is.

```
jshell> int egyenleg = 0
egyenleg ==> 0

jshell> void bevetel(int osszeg) {
...>     egyenleg = egyenleg + osszeg;
...> }
| created method bevetel(int)

jshell> bevetel(23)

jshell> bevetel(78)

jshell> bevetel(-34)

jshell> egyenleg
egyenleg ==> 67
```

Amikor egy függvénynek nincs közvetlen visszatérési értéke, akkor azt úgy jelöljük, hogy „void” típusúként határozzuk meg a kimenetet. Ilyenkor a függvény definíciójában nem kell „return” záradék. Természetesen ha a függvény eredménye nem void, akkor kell „return”.

void függvényeket értelemszerűen olyan esetekben használunk, amikor a függvény a tevékenységét az általa elérhető környezeti változókon hajtja végre, magyarul valamit a környezetében módosít, megváltoztat.

(A gyakorlatban a függvények túlnyomó többsége osztályokba, objektumokba ágyazottan működik, ahol a környezeti változói az objektum példány belső változói.)

Ahogy nem kötelező, hogy a függvénynek legyen közvetlen visszatérési értéke, úgy az sem kötelező, hogy bemeneti paramétere legyen. Ilyenkor a függvény kizárólag az általa elérhető környezeti változókból gazdálkodik.

Példa:

```
jshell> double kamatlab = 6.4; int toke = 34223
kamatlab ==> 6.4
toke ==> 34223

jshell> double kamat() { return toke * kamatlab / 100; }
| created method kamat()

jshell> kamat()
$4 ==> 2190.272

jshell> toke = 3
toke ==> 3

jshell> kamat()
$6 ==> 0.19200000000000003
```

A függvényeknek bárhány bementi paraméterük lehet, visszatérési értékük azonban legfeljebb egy. Ez azonban lehet összetett adatstruktúra is (objektum, tömb, lista stb.).

## Feladatok

1. Írj függvényt, ami kiszámítja két bemeneti egész szám átlagát.
2. Írj függvényt az általános másodfokú polinom kiszámítására.

$$f(x) = ax^2 + bx + c$$

Az együtthatók (a, b, c) legyenek környezeti változók, a függvény közvetlen bemeneti paramétere csak az x legyen.

3. Írj függvényeket a derékszögű háromszög átfogójának, területének és kerületének kiszámításra. A bemeneti paraméterek minden esetben a háromszög két befogója.
4. Környezeti változóban tárold az euró aktuális árfolyamát és írd mellé két függvényt. Az egyik forintot számol át euróba, a másik meg fordítva.
5. Írj függvényt a kamatos kamat kiszámítására. A kamatláb továbbra is legyen környezeti változó, bemeneti paraméter pedig a kezdeti tőke és az évek száma.